GENERALIZED GROUPING AND LOADING PROBLEMS IN FMS : MODELS AND METHODOLOGIES

A Thesis Submitted
in Partial Fulfilment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

by

ANIL KUMAR AGRAWAL

to the

INDUSTRIAL AND MANAGEMENT ENGINEERING PROGRAMME

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

AUGUST, 1990

IME-1990-D-AGR-GEN

CERTIFICATE

It is certified that the work contained in the thesis entitled "GENERALIZED GROUPING AND LOADING PROBLEMS IN FMS : MODELS AND METHODOLOGIES," by "Anil Kumar Agrawal," has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

(Kripa Shanker)
Prof and Head
Industrial and Management Engineering
I.I.T. Kanpur

August, 1990

ABSTRACT/SYNOPSIS

Name of Student Anil Kumar Agrawal Roll No. 8611461

Degree for which submitted Ph.D. Department Industrial and Management Engineering

Thesis Title GENERALIZED GROUPING AND LOADING PROBLEMS IN FMS: MODELS AND METHODOLOGIES

Name of Supervisor Kripa Shanker

Month and year of thesis submission August, 1990

The present thesis deals with grouping and loading problems in Flexible Manufacturing Systems (FMS) possessing routing and tooling flexibilities where operations can be performed with more than one combination of machines and tool types, leading to alternative process plans for the parts. Models encompassing various design and operational problems related to machine and part grouping, and resource allocation, have been presented and associated solution methodologies are suggested.

For the grouping purpose in the generalized situation, two new measures of commonality, Relative Requirement Compatibility (RRC) and Absolute Requirement Compatibility, are introduced. The measures use the bases of processing times, the number of operations, and the number of machines and tools, for determining the closeness between a pair of parts, process plans, and a part (process plan) and a machine. Further, the relationships between RRC, Jaccard's similarity coefficient, product type similarity

coefficient, and cell bond strength are established and analyzed to investigate their relative discriminating characteristics. It is found that the RRC is better than the others. Illustrations presented confirm this observation, and also show the RRC to be better than the simple matching similarity coefficient. An analysis on the values of these commonality measures is also carried out to show their comparative discriminating power. In addition, the analysis provides an insight for determining the threshold values for these measures which are needed by certain grouping algorithms and methodologies. Numerical examples solved using p-median problem formulation approach further show that the RRC, as compared to the other measures, provides better results.

For generalized grouping problem, models have been developed using the p-median, p-centre and graph partitioning problem formulation approaches.

The existing p-median formulation of Kusiak and equivalent generalized assignment problem formulation of Shtub are investigated and improved formulations requiring lesser number of variables and constraints are presented. The shortcoming of these models of sometimes not being able to yield the perfect groups even when they exist, is rectified by incorporating an additional constraint on group disjointedness. This problem, to some extent, is also reduced by using RRC instead of the simple matching similarity measure used by these authors. In addition, the models using RRC are found to encompass more generality and flexibility of the production system.

The grouping problem is extended to consider some additional factors: limits on the size of the part families and machine cells

the number of machines of each type, and the capacity of the machines. It is shown that the consideration of machine capacity is quite important in case of generalized grouping, and provides solution to the loading problem also.

In p-median problem formulation approach, the objective is to maximize the sum of the similarity between the member process plans and the plans representing the corresponding process families. The approach of p-centre formulation, however, maximizes the minimum similarity of the member process plans with the plans representing the corresponding part families.

Next, the generalized grouping problem is formulated using the approach of graph partitioning. The mathematical models presented are shown to be better than the equivalent p-median problem formulations and its extensions in terms of the required number of constraints and variables. However, for the simple grouping problem where each part has a single process plan, the graph partitioning approach is found not to be very efficient.

Models have also been developed to integrate the generalized grouping problem with the cellular production system design problem of determining the machine cells requiring the minimum investment.

The models developed for the grouping problems are observed to be either NP-complete or hard to solve. Therefore, for some of these problems, heuristic approaches have also been proposed. In the heuristics proposed for p-median and p-centre problem, p process plans, each belonging to a different part, are selected first in a greedy manner to represent p process families, and then process plans, one each of the remaining parts, are assigned to

these p plans. Improvements are brought in the objective function value by perturbing selectively the plans representing the process families. In graph-partitioning approach, the formation of part families and machine cells is carried out simultaneously with the objective of maximizing the association between parts and machines belonging to the same group. Besides these heuristics, for some of the models based on p-median and graph-partitioning formulations, Lagrangian relaxation approach has been proposed where certain complicating constraints are relaxed and the resulting expressions are found to have special structure that can be solved using some standard procedures. It has also been observed that for some of the models, the Lagrangian relaxation approach may not serve the purpose of reducing the computational complexity. Numerical examples have been presented to illustrate the use of some of these models and heuristics.

It has been shown that similarity coefficients that attempt to find similarity between a pair of the same entities, such as, parts, machines and process plans, play important role in hierarchical grouping where each part has a single process plan. Whereas, in generalized grouping, it is not of much use. Further, the graph-partitioning approach is shown not to be of much use in the case of hierarchical group formation.

An important feature of the grouping models is that while deciding the groups, the solution is also obtained for the assignments of operations to the various machines. However, some graph-partitioning based models do not find a complete solution to this problem. For this purpose, the loading models developed can be used for finding a complete solution to operations assignment

problem. These loading models, in addition, also provide solutions to tool allocation problems.

The salient feature of the loading models is that they do not contain nonlinear terms to represent savings in the total slot requirement due to tool duplication and overlapping. The savings due to tool duplication are taken care of by the formulation itself, while explicit expressions for the savings due to overlaps are avoided by using a scheme developed for counting the slot requirements of the tools.

Generally, incorporation of flexibility into consideration makes the loading problem complex to solve and analyze, but helps to increase the throughput rate and to utilize the resources in a better manner. With this in view, certain quantitative measures for the flexibility of processing a part have been proposed, and loading models are presented incorporating such measures. Numerical examples presented exhibit the advantages achieved.

ACKNOWLEDGEMENTS

It is a matter of great pride and distinct priviledge to have been associated with individual like Dr. Kripa Shanker, Professor and Head, Industrial and Management Engineering Programme, Indian Institute of Technology, Kanpur, who not only provided encouraging guidance but also exerted personally till late hours in the night discussing various aspects of the thesis both in terms of its content and quality. I, therefore, remain to be indebted to him for his immeasurable support, deliberations and suggestions. I am also thankful for the meticulous care he has taken to improve the presentation of the material in this thesis.

Special thanks are due to Mrs. Kripa Shanker whose innumerable kindness can never be forgotten. Her patience is worth to be accorded. I am grateful to her for the cups of tea she provided at odd hours that helped in stimulating discussions.

I express my sincere thanks to the entire faculty of the Department of Industrial and Management Engineering who provided me all possible help and support during my stay at this Institute. Special mentions are for Prof. S. Sadagopan and Prof. A.K. Mittal of Industrial and Management Engineering Programme, and Prof. S.K. Gupta, Professor in the Department of Mathematics.

I gratefully acknowledge the kind help that I received from my fellow research scholar, T. Sattanathan, and other M.Tech. scholars, Modi, Sukhram Pal, Dangi and Murthy.

I express my gratitude to the Ministry of Human Resource Development, Department of Education, Government of India, for the

financial support extended through the Quality Improvement Programme, and the Banaras Hindu University for relieving me of duties during the tenure of this work.

I also express my appreciation to Sri J. K. Mishra, Sri G. L. Mishra, Sri P. D. Gupta and Miss Sandhya for excellent typing, and Sri G. K. Shukla for his patience for drawing the figures.

I would also like to thank the staff members of CAM project, in particular, Mr. O. P. Bajaj who extended all possible support for the completion of the present work. Thanks are also due to the staff members of Industrial and Management Engineering and also of Q.I.P. office who helped me a lot in all the official proceedings.

In the last, I would like to thank all those who have made my stay in I.I.T. premises pleasant and memorable one.

August, 1990 (Anil Kumar Agrawal)

TABLE OF CONTENTS

	Page No.
CHAPTER I : INTRODUCTION	1
1.1 GT: AN OVERVIEW	6
1.2 GT AND FMS	12
1.3 METHODOLOGIES FOR GT CELL FORMATION	14
1.3.1 Production/Component Flow Analysis	14
1.3.2 Coding and Classification Systems	15
1.3.3 Expert Systems	16
1.3.4 Graph-Theoretic Approaches	17
1.3.5 Optimization Techniques	19
1.3.6 Clustering	20
1.3.6.1 Similarity Coefficient Based Approaches	21
1.3.6.2 Array Based Approaches	24
1.3.6.3 Other Heuristics	25
1.4 GT AND OTHER FUNCTIONS	27
1.4.1 Material Requirements Planning	28
1.4.2 Loading	29
1.4.3 Scheduling and Sequencing	30
1.5 RESEARCH ISSUES	31
1.6 PRESENT WORK	32
1.6.1 Scope	32
1.6.2 Organization	33

2.2.2	For a Pair of Parts	55
2.2.3	For a Pair of a Process Plan and a Machine	56
2.2.3.1	BASED ON THE PROCESSING TIMES	57
2.2.3.2	BASED ON THE NUMBER OF OPERATIONS	57
2.2.3.3	BASED ON NUMBER OF TOOLS	57
2.2.3.4	SOME COMMENTS AND OBSERVATIONS	58
2.3	RELATIONSHIP OF RELATIVE REQUIREMENT COMPATIBILITY WITH OTHER COMMONALITY MEASURES FOR A PAIR OF PART	59
2.3.1	Mathematical Relationship	59
2.3.2	Observations on Interrelationship and Definitions of Commonality Measures	62
2.4	COMPUTATIONAL REQUIREMENT	65
2.5	DISCRIMINATING POWER	66
2.6	COMMONALITY MEASURES AND THEIR NUMERICAL VALUES	74
2.6.1	Sensitivity of Values for Capturing Variations in Problem Scenarios	88
2.6.2	Nature of Variations in the Values of Commonality Measures	89
2.6.3	Value of Commonality Measure Versus Degree of Closeness	92
2.6.3.1	SAME THRESHOLD VALUE FOR EACH COMMONALITY MEASURE	94
2.6.3.2	DIFFERENT THRESHOLD VALUES FOR COMMONALITY MEASURES	95
2.7	NUMERICAL EXAMPLES	103
2.8	SUMMARY AND CONCLUSIONS	115
CHAPTER III :	GENERALIZED GROUPING PROBLEMS: BASIC PROBLEM AS A P-MEDIAN PROBLEM	118
3.1	INTRODUCTION	118
3.2	THE FORMULATIONS OF KUSIAK AND SHTUB	123
3.2.1	Kusiak's Formulation	125
3.2.2	Shtub's Formulation	127
3.3	OBSERVATIONS ON THE FORMULATIONS OF KUSIAK AND SHTUB	135
3.3.1	Grouping Approach	135
3.3.2	Formulation	136

3.3.2.1 GROUP DISJOINTEDNESS 136
3.3.2.2 COMPUTATION OF SIMILARITY 138

3.3.2.3 PROCESS PLAN ASSIGNMENT CONSTRAINT 139
3.3.2.4 CONSIDERATION OF MACHINE CAPACITY 140

3.3.3 Input Requirement 140

3.3.3.1 OPERATION AGGREGATION 141
3.3.3.2 AGGREGATION OF MULTIPLE COPIES OF A MACHINE TYPE 141

3.4 IMPROVEMENTS IN THE FORMULATIONS OF KUSIAK AND SHTUB 142

3.4.1 Improvements in Kusiak's Formulation 142
3.4.2 Improvements in Shtub's Formulation 147

3.5. EXTENSIONS 152

3.5.1 Machines Denoted by Their Types 158

3.5.1.1 IDENTICAL MACHINES ASSIGNED TO A SINGLE GROUP 158
3.5.1.2 IDENTICAL MACHINES DISTRIBUTED AMONG GROUPS 166

3.5.2 Machines Considered Separately 174

3.5.2.1 INDICES OF MACHINES LISTED AGAINST EACH OPERATION 175
3.5.2.2 INDICES OF MACHINE TYPES LISTED AGAINST EACH OPERATION 181

3.6. SOLUTION METHODOLOGY 185

3.6.1 Heuristic for the Maximization of the Total Relative Requirement Compatibility 186

3.6.1.1 PROCEDURE 187
3.6.1.2 SOLUTION METHODOLOGY FOR MATHEMATICAL MODELS APPEARING IN THE HEURISTIC 191
3.6.1.3 DISCUSSION ON VARIOUS STEPS OF THE HEURISTIC 192
3.6.1.4 COMPUTATIONAL COMPLEXITY 193
3.6.1.5 EXAMPLES 194

3.6.2 Generalizations of the Heuristic 200

3.6.2.1 CONSTRAINT ON THE MAXIMUM NUMBER OF PARTS IN A GROUP 201
3.6.2.2 CONSTRAINT ON THE MINIMUM NUMBER OF PARTS IN A GROUP 202

3.6.2.3 CONSTRAINTS ON THE MINIMUM AND MAXIMUM NUMBER OF PARTS IN A GROUP 207

3.6.3 Lagrangian Relaxation Method 209

3.6.3.1 DETERMINATION OF PART FAMILY ONLY 210
3.6.3.2 SIMULTANEOUS DETERMINATION OF GROUPS OF PARTS AND MACHINES 219

3.6.4 Some Comments 222

3.6.4.1 INCORPORATION OF OTHER COMMONALITY MEASURES IN THE HEURISTIC 222
3.6.4.2 GROUPING APPROACH 222

3.7 NUMERICAL EXAMPLES 226
3.8 SUMMARY AND CONCLUSIONS 236

CHAPTER IV : GENERALIZED GROUPING PROBLEMS: BASIC PROBLEM AS A P-CENTRE PROBLEM 238

4.1 INTRODUCTION 238
4.2 PROBLEM STATEMENT 239
4.3 SOLUTION METHODOLOGY 240

4.3.1 Heuristic for Maximizing the Minimum Relative Requirement Compatibility 240
4.3.2 Method for Solving Mathematical Models in the Heuristic 245
4.3.3 Comments on Various Steps of the Heuristic 246
4.3.4 Computational Complexity 247

4.4 EXAMPLES 247
4.5 SUMMARY AND CONCLUSIONS 252

CHAPTER V : GENERALIZED GROUPING PROBLEMS: BASIC PROBLEM AS A GRAPH PARTITIONING PROBLEM 253

5.1 INTRODUCTION 253
5.2 PROBLEM ENVIRONMENT 259
5.3 GROUPING APPROACH 261
5.4 GROUPING AS AN OPERATIONAL PROBLEM 267

5.4.1 Formulation 267
5.4.2 Solution Methodology 278

5.4.2.1 DETERMINATION OF PART FAMILY ONLY 278

5.4.2.2 SIMULTANEOUS DETERMINATION OF GROUPS WITHOUT CONSIDERING MACHINE CAPACITY 279

5.4.2.3 SIMULTANEOUS DETERMINATION OF GROUPS CONSIDERING MACHINE CAPACITY 286

5.4.3 An Alternative Approach for Grouping 290

5.4.3.1 GENERALIZED GROUPING PROBLEM 290

5.4.3.2 SIMPLE GROUPING PROBLEM 293

5.5 GROUPING AS A PLANNING PROBLEM 311

5.5.1 Formulation 312

5.5.2 Solution Methodology 313

5.6 NUMERICAL EXAMPLES 313

5.7 SUMMARY AND CONCLUSIONS 323

CHAPTER VI : LOADING PROBLEMS IN FMS 325

6.1 INTRODUCTION 325

6.2 PROBLEM ENVIRONMENT 338

6.3 LOADING MODELS FOR UNIQUE ROUTING AND TOOLING 340

6.3.1 All Machines Versatile 341

6.3.1.1 MINIMIZATION OF PROCESSING COST 341

6.3.1.2 MINIMIZATION OF MAXIMUM LOAD ON MACHINES 351

6.3.1.3 MINIMIZATION OF THE SUM OF OVERLOADS ON MACHINES 355

6.3.2 Machines not Necessarily Versatile 356

6.3.2.1 MINIMIZATION OF PROCESSING COST 357

6.3.2.2 MINIMIZATION OF MAXIMUM WORKLOAD ON MACHINES 362

6.3.2.3 MINIMIZATION OF THE SUM OF OVERLOADS ON MACHINES 363

6.4 LOADING MODELS WITH ALTERNATE ROUTING 365

6.4.1 Operations not Duplicated 366

6.4.2 Duplicate Assignments of Operations 372

6.5 SUMMARY AND CONCLUSIONS 379

CHAPTER VII : SUMMARY AND CONCLUSIONS 381

REFERENCES 387

BIBOLIOGRAPHY 395

LIST OF TABLES

Table No.	Particulars	Page No.
2.1	Commonality Measure Values for Different Problem Scenarios for the Comparison of the Discriminating Characteristics.	67
2.2	Blocks of Scenarios Given in Table 2.1.	68
2.3	Distributions for Matching of Machine Requirements.	76
2.4	Some Related Statistics on Various Commonality Measures for Different Matching Requirement Distribution	87
2.5	Solution to Example 2.1 for p = 2.	105
2.6	Solution to Example 2.2 for p = 2.	106
2.7	Solution to Example 2.2 for p = 3.	108
2.8	Solution to Example 2.3 for p = 2.	108
2.9	Solution to Example 2.4 for p = 2.	109
2.10	Solution to Example 2.5 for p = 2.	110
2.11	Solution to Example 2.5 for p = 3.	111
2.12	Solution to Example 2.5 for p = 4.	111

3.1	Model M3.2 and Shtub's Matrix Representation of Generalized Grouping Problem.	134
3.2	Data For Example 3.1.	137
3.3	Number of Constraints and Variables for Different Models.	146
3.4	Correspondence between model M3.4 and matrix representation given in Figure 3.3.	150
3.5	Data For Example 3.2.	195
3.6	Data For Example 3.3.	198
3.6(a)	Value of Certain Parameters (Example 3.4).	226
3.6(b)	Details of Machines (Example 3.4).	227
3.6(c)	Details of Process Plans for Various Parts (Example 3.4).	227
3.7	The Detail of Problem Scenarios Considered for Example 3.4.	228
3.8	Grouping Solution For Different Problem Scenarios for Example 3.4.	229
3.9	Load on Machines for the Grouping Shown in Table 3.8 (Example 3.4).	229
3.10(a)	Values of Certain Parameters (Example 3.5).	231
3.10(b)	Details of Machines (Example 3.5).	231

3.10(c)	Details of Process Plans for Various Parts (Example 3.5).	232
3.11	Grouping Solution for Example 3.4 Considering Process plans.	233
3.12	Details of Route Plans for Various Parts (Example 3.5)	235
3.13	Solution to Example 3.5 Considering Route Plans.	236
4.1	Solution for the Example Shown in the Figure 4.2.	251
5.1	Operations Assignment Detail (Example 5.2).	267
5.2	Number of Constraints and Variables Required by the Formulations for Determination of Only Part Family.	274
5.3	A Comparison Between Requirements of Decisions Variables and Constraints for Examples 5.1 and 5.2 for the Formulations Based on p-median and Graph Partitioning.	274
5.4	Number of Constraints and Variables Required by the Formulations for Simultaneous Determination of Part Families and Machine Cells	276
5.5	Processing Requirement and Productivity Potential Details of the Operations of the Various Parts Associated with the Compatible Machines.	314
5.6	Details of the Machines.	315

5.7	Operations Assignment Detail for the Case of Only Part Family Determination.	315
5.8	Operation Assignment Details for the Case When the Operations are Carried Out Within the Cell and Machine Capacity Constraint is not Considered.	316
5.9	Group Configuration Details for the Case When the Operations are Carried Out Within the Cell and Machine Capacity Constraints is not Considered.	317
5.10	Operation Assignment Details for the Case When the Operations are Carried Out Within the Cell and Machine Capacity Constraint is Included.	318
5.11	Operation Assignment Details for the Case When Machine Capacity Constraint is not Included and the Restriction on Group Disjointedness is Relaxed.	319
5.12	Group Configuration Details for the Case When Machine Capacity Constraint is not Included and the Restrictions on Group Disjointedness is Relaxed.	319
5.13	Operation Assignment Details for the Case When Machine Capacity Constraint is Included and the Restriction on Group Disjointedness is Relaxed.	320
5.14	Group Configuration Details for the Case When Machine Capacity Constraint is Included and the Restriction on Group Disjointedness is Relaxed.	320

5.15	Operation Assignment Details for the Case When Groups are Determined Minimizing Investment on Machines.	321
5.16	Group Configuration Details for the Case When Groups are Determined Minimizing Investment on Machines.	321
5.17	Operation Assignment Obtained Using the Heuristic GAO.	322
6.1	Resource Requirements and Related Costs for the Operations of the Parts (Example 6.1).	352
6.2	Available Copies and Slot Requirements of Various Tool Types (Example 6.1).	353
6.3	Machine and Tool Magazine Capacity (Example 6.1).	353
6.4	Operations and Tools Assignments for the Objective of Minimizing Total Processing Cost (Example 6.1).	353
6.5	Tool Allocation and Capacity Utilization for the Objective of Minimizing the Total Processing Cost (Example 6.1).	354
6.6	Operation Assignment Detail for the Objective of Minimizing the Maximum Workload on Machines (Example 6.1).	354
6.7	Tool Allocation and Capacity Utilization for the Objective of Minimizing the Maximum Workload on Machines (Example 6.1).	355

6.8	Resource Requirements and Related Costs for the Operations on the Parts (Example 6.2).	360
6.9	Available Copies and Slot Requirements of Various Tool Types (Example 6.2).	361
6.10	Machine and Tool Magazine Capacity, and Compatibility Details (Example 6.2).	361
6.11	Operation Assignment Details for the Objective of Minimizing the Total Processing Cost (Example 6.2).	362
6.12	Tool Allocation and Capacity Utilization for the Objective of Minimizing the Total Processing Cost (Example 6.2).	362
6.13	Operation Assignment Details for the Objective of Minimizing the Maximum Workload on Machines (Example 6.2).	364
6.14	Tool Allocation and Capacity Utilization Detail for the Objective of Minimizing the Maximum Workload on Machines (Example 6.2).	364
6.15	Resource Requirements and Related Costs for the Operations on the Parts (Example 6.3).	369
6.16	Available Copies and Slot Requirements of Various Tool Types (Example 6.3).	370
6.17	Machine and Tool Magazine Capacity, and Compatibility Details (Example 6.3).	370
6.18	Operation Assignment Detail for the Objective of Minimizing the Total Processing Cost (Example 6.3).	371

6.19 Tool Allocation and Capacity Utilization for the Objective of Minimizing the Total Processing Cost (Example 6.3). 371

6.20 Operation Assignment Detail for the Objective of Minimizing the Maximum Workload on Machines (Example 6.3). 373

6.21 Tool Allocation and Capacity Utilization for the Objective of Minimizing the Maximum Workload on Machines (Example 6.3). 373

6.22 Operation Assignment Detail for the Objective of Maximizing the Part Processing Flexibility (Example 6.3). 378

LIST OF FIGURES

Figure No.	Particulars	Page No.
2.1	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Normal (μ = 0.25, σ = 0.0888).	77
2.2	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta (α = 1.1, β = 3.3).	78
2.3	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta (α = 0.5, β = 1.0).	79
2.4	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta (α = 0.9, β = 0.9).	80
2.5	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Uniform (a = 0.0, b = 1.0).	81
2.6	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Triangular (Symmetric with mean = 0.5).	82
2.7	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Normal (μ = 0.5, σ = 0.1666).	83

2.8	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta (α = 1.0, β = 0.5).	84
2.9	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta (α = 3.3, β = 1.1).	85
2.10	Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta (α = 5.5, β = 1.1).	86
2.11	Relationship Between the Ratio of Interval Length and Standard Deviation (R/σ) for RRC and CBS.	90
2.12	Relationship Between the Mean Values of RRC and PSC.	90
2.13	Proportional of Additional Pairs for Which at least onf of the RRC for a Pair of Parts is not Less Than the Threshold Limit ($=T_{RRC}$).	98
2.14	Proportion of the Cases Missed by One and Acknowledged by the Other Measure (Measures: RRC and CBS).	99
2.15	Proportion of the Cases Missed by One and Acknowledged by the Other Measure (Measures: RRC and JSC).	100
2.16	Proportion of the Cases Missed by One and Acknowledged by the Other Measure (Measures: RRC and PSC).	101

2.17	Example 2.1 (Seifoddini (1989c), original matrix is transposed).	105
2.18	Example 2.2 (Kumar et al (1986)).	106
2.19	Example 2.3 (King and Nakornchai (1982)).	107
2.20	Example 2.4 (Chandrashekharan and Rajagopalan (1986), original matrix is transposed).	109
2.21	Example 2.5.	110
2.22	Relative Requirement Compatibility Matrix $[RRC_{ij}]$ for Example 2.5.	113
2.23	Cell Bond Strength Matrix $[CBS_{ij}]$ for Example 2.5.	114
2.24	Product Type Similarity Coefficient Matrix $[PSC_{ij}]$ for Example 2.5.	114
2.25	Jaccard's Similarity Coefficient Matrix $[JSC_{ij}]$ for Example 2.5.	114
3.1	The Table Representation of GAP.	128
3.2	Matrix Representation of the Generalized Grouping Problem Formulated as GAP by Shtub (1989) (* in a cell corresponds to $R_{ij} = -\infty$ and $\gamma_{ij} = \infty$).	129
3.3	Matrix Representation of the Grouping Problem Corresponding to Model M3.4.	149

3.4	Flowchart for the Heuristic for Determining the Groups While Maximizing the Total of Similarity Value.	188
3.5	The Relative Requirement Compatibility Matrix $[R_{ij}]$ For Example 3.2.	196
3.6	The Relative Requirement Compatibility Matrix $[R_{ij}]$ For Example 3.3.	198
3.7	Transportation Matrix Corresponding to the Problem P3.1 with Restriction on The Minimum of Number Parts in Each Group (* in a cell represents a value of $-\infty$).	206
4.1	Flowchart for the Heuristic for Determining the Groups While Maximizing the Minimum of the Similarity Value.	242
4.2	Relative Requirement Compatibility Matrix $[R_{ij}]$.	251
5.1(a)	Initial Part-Machine Matrix (Example 5.1).	262
5.1(b)	Corresponding Bipartite Graph (Example 5.1).	262
5.2(a)	Rearranged Part-Machine Matrix (Example 5.1).	263
5.2(b)	Rearranged Bipartite Graph (Example 5.1).	263
5.3	AND-OR Graph Representation for a Generalized Grouping Situation (Example 5.2).	265
5.4	Decomposed AND-OR Graph (Example 5.2).	266

CHAPTER V

A GENERALIZED GT APPROACH : BASIC PROBLEM AS A GRAPH PARTITIONING PROBLEM

5.1 INTRODUCTION

The approaches employed for determination of groups of parts and machines are either hierarchical or non-hierarchical in nature. In hierarchical approaches, the groups of parts and machines are determined sequentially. If first machine groups are determined, then next machine cells will be determined. Similarly, when first part families are determined, then next machine cells will be determined. In case of non-hierarchical grouping, the part families and machine cells are determined simultaneously.

In simple grouping situations where each part has fixed routing, it is difficult to say , in general, as which of the two approaches, viz. hierarchical and non-hierarchical, will require lesser computational effort and yield better grouping. The characteristics of the groups formed depend largely upon the methodology chosen for group formation. However, in generalized grouping situations where an operation can be performed on more than one machine, simultaneous determination of groups is not an easy task and is expected to take more computational effort as compared to when groups are determined hierarchically. This aspect has been elaborated in the previous chapter, and it is this reason because of which a hierarchical approach has been suggested for dealing with the generalized grouping problem.

5.5	Transportation Matrix Corresponding to Subproblem P_b.	283
5.6	Flowchart Corresponding to Phase I of the Heuristic for the Generalized Grouping.	294
5.7	Flowchart Corresponding to Phase II of the the Heuristic for the Generalized Grouping.	295
5.8	Flowchart Corresponding to Phase I of the the Heuristic for the Simple Grouping.	304
5.9	Flowchart for the Phase II of the Heuristic for the Simple Grouping Problem.	305
5.10	Flowchart of the Algorithm Used for Finding Disconnected Components of a Graph.	307
5.11	Flowchart Showing the Method Used for Obtaining Given Number of Groups in the Simple Grouping Situations When the Total Number of Groups Formed are More than What is Required.	308
5.12	Flowchart Showing Details for Creating the Required Number of Additional Groups in Simple Grouping Case.	309
5.13	Flowchart Showing Details for Rearrangement of Parts and Machines Among the Groups for Satisfying the Limits on Group Size.	310
6.1	Tool Drum with Tools Mounted on it.	343
6.2	An Example for Illustrating Tool Slot Saving.	345

arise because of the separate consideration given to the number of parts and machines that are to be assigned to the various part families and machine cells.

5.2 PROBLEM ENVIRONMENT

The grouping problem envisaged in the present chapter considers the production environment which is better described by the following.

(i) All the machines of the same characteristic and capability are considered to be of the same type and thus there may be more than one machine of the same type.

(ii) The parts have multiple operations.

(iii) An operations can be performed on more than one machine type and related processing costs and times may be different.

(iv) The number of groups to be formed is prespecified.

The points mentioned above relate to a general production situation, such as FMS, where the machines are more flexible and are capable of performing different operation types.

Though the production system described above is flexible and allows for alternate routing of parts, but the groups ultimately determined restrict the parts to follow only one route.

The models for the grouping problems are described in the subsequent sections. For the development of the formulation following notations are used.

Indices

k : a group

n : a part

i : an operation of a part

CHAPTER I

INTRODUCTION

The present trend of increasing complexities in the product design and the requirements of higher reliability and performance characteristics has resulted into the development of newer materials, automated and sophisticated machine tools, and advanced processing technology in the manufacturing industry. This composite and complex phenomenon along with competitive marketing environment is responsible for decreasing product life cycle. This implies that a manufacturing system should be equipped to produce a variety of products with least possible lead time at a minimum possible cost. Flexible automation with capability to accommodate such dynamic parameters of the emerging manufacturing and marketing scenario has proved to be a solution for satisfying such requirements. The advent of computer technology has played a very important role in automation. Further, a high capital investment associated with automation demands more efficient system utilization that can only justify this with a matching return on investment and a high productivity. This in turn, necessitates an efficient decision making for planning and control, and integration of various functions of manufacturing.

One of the recent developments to satisfy the above complex requirements regarding machine tool automation and integrated decision making is Flexible Manufacturing System (FMS).

The aim of an FMS is to achieve the efficiency of automated high volume mass production while retaining the flexibility of low

to the following:

$$\text{Maximize} \sum_{n=1}^{N} \sum_{i\in B(n)} a_{ni} \sum_{m\in C(i)} \sum_{k=1}^{p} x_{nimk}. \tag{5.2}$$

It can be seen that the expression (5.2) representing the objective of minimizing the intercell movement cost is similar to the expression (5.1) representing the objective of maximizing the sum of associations between parts and machines belonging to the same group.

It should be noted that the intercell movement cost expressed above does not consider the technological sequence among the operations, layout of machines and scheduling rules to be used, and thus may not reflect actual cost. Since the operation assignment problem is being solved alongwith the grouping problem, determination of actual intercell movement cost would otherwise be not possible. Actual cost can be more or less as compared to that expressed above. Expression for the intercell movement cost given above can be understood to reflect the average behaviour and thus to be a good measure.

CONSTRAINTS

The various constraints are as follows.

(i) <u>Assignments of Operations</u>

An operation of a part is assigned to one and only one machine that belongs to a group to which the part is assigned. The constraint is expressed as:

$$\sum_{m\in C(i)} x_{nimk} = y_{nk} \qquad n = 1, \ldots, N;\ \forall\ i \in B(n)$$
$$k = 1, \ldots, p. \tag{5.3}$$

Thus, the groups determined in this way may be highly disjoint or for the complete independence these may require good number of duplicate machines. On the other hand, the methodologies which use similarity coefficients, such as p-median formulation, will try to put in a group those parts that are close in terms of their machine requirements and the groups formed will be less disjoint.

It can be concluded from the above that for a better group formation, the hierarchical approaches should use some kind of similarity coefficient.

5.4.2.2 SIMULTANEOUS DETERMINATION OF GROUPS WITHOUT CONSIDERING MACHINE CAPACITY

It has been stated by Kumar et al (1986) that the graph partitioning formulation for the simple grouping problem is NP-complete. Since an instance of the generalized grouping problem (model M5.2) itself will represent simple grouping problem, thus the problem formulation represented by the models M5.2 and M5.3 will also be NP-complete. As mentioned earlier, mathematical programming techniques can anyway be used for solving such problems. However, because of the NP-complete structure of the problem, it may not be computationally efficient.

For the reason mentioned above, a heuristic approach based on Lagrangian relaxation is proposed for the problem of simultaneous group determination where machine capacity constraint is not considered and the constraint on assignment of all the operations is relaxed (Model M5.4). The relaxed problem after dualizing the constraint (5.5) can be written as follows.

$$Z_D(w) = \text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} a_{nim} \sum_{k=1}^{p} x_{nimk}$$

volume job shop production. The conflicting objectives of the high productivity and high flexibility in FMS are achieved through appropriate selection and layout of the equipment and facilities (viz. machines, tools, jigs, fixtures, material handling system, etc.), computer software and hardware, and integrated decision making system both for long term design decisions as well as for short term operational decisions.

An FMS, in general, consists of a group of processing stations - numerically controlled machines, machining centers with automatic tool interchange capabilities and robots-linked together with an automatic materials handling system and automatic storage and retrieval system that operate as an integrated system under the control of a central computer.

The flexibility of an FMS is attributed mainly to the multifunctionality and programmability of machines and material handling equipment. The feedback control system and the central computer are, of course, the main integrating factors. Combinations of these flexible components of FMS lead to various kinds of flexibility. Some of the popular measures of flexibility (Kusiak (1985), Yao (1985), Stecke and Browne (1985)) are as follows:

Measure of Flexibility	Description
Machine Flexibility	Ability of a machine to perform a variety of tasks (multi-functionality of machine).
Process Flexibility	Ability to produce a given set of part types in several ways using different combinations of machines and cutting tools. It measures the ability of the system to employ more than one type of tool and machine for

(i) Set $\underline{x}_{nimk} = x_{nimk}$ for $n = 1, \ldots, N$; $\forall\, i \in B(n)$; $\forall\, m \in C(i)$ and $k = 1, \ldots, p$.

(ii) Find for all the machine types $m \in G_M(k)$ load of those operations of parts (i.e. $\forall\, n \in G_p(k)$) for which $\underline{x}_{nimk} = 1$.

(iii) If load on any machine type $m \in G_M(k)$ is more than $T_m \cdot Z_{mk}$, then in the decreasing order of the value of $(a_{nim} | t_{nim})$ the operation i of part $n \in G_p(k)$ is removed from machine type m. Set $\underline{x}_{nimk} = 0\ \forall\, n \in G_p(k)$ and $\forall\, i \in B(n)$ if $m \in C(i)$ but $m \notin G_M(k)$.

(iv) Remove from consideration those parts whose all the operations have been assigned and also the operations of remaining parts that have already been assigned. Remove from consideration all those machines whose capacities have been used completely.

(v) Let the set of remaining parts and machines be $G_p'(k)$ and $G_M'(k)$ and the set of unassigned operations of part $n \in G_p'(k)$ be $B'(n)$ and that for the machine $m \in G_M'(k)$ be $\underline{B}'(m)$. Let the remaining capacity of all the machines of the type $m \in G_M'(k)$ be T_m'.

The generalized assignment problem to be solved now will be quite small in size as compared to that given before. The problem is to:

Maximize $$\sum_{m \in G_p(k)} \sum_{i \in B(n)} \sum_{m \in C(i) \cap G_M(k)} a_{nim} \cdot \underline{x}_{nimk}$$

Subject to:

$$\sum_{m \in C(i) \cap G_M'(k)} \underline{x}_{nimk} \le 1 \qquad \forall\, n \in G_p'(k); \quad \forall\, i \in B'(n)$$

	carrying out an operation.
Volume Flexibility	Ability to operate the system profitably at different production volumes.
Expansion Flexibility	Ability to expand the production capacity easily and economically with minimum interlocking.
Tooling Flexibility	Ability to employ more than one tool (of the same or different type) for an operation and also the ability of a tool to perform several operations on different machines.
Materials Handling Flexibility	Ability to handle different parts on a number of different routes.
Product Flexibility	Ability to changeover to produce a new set of products economically and quickly.
Production Flexibility (Job Flexibility)	Extent or size of the set of part types that the system is able to produce. It measures the ease of making changes in machine setup for parts with varying design and manufacturing characteristics.
Operation Flexibility	Ability to interchange the order of certain operations of parts.
Routing Flexibility (Scheduling Flexibility)	Ability to have alternate routing of parts (due to multifunctionality of machines).

Some of the above measures of flexibility are interdependent in their connotation. For example, process and routing flexibilities are similar to each other in terms of their basic characteristics. A system with more process flexibility will obviously have greater amount of routing flexibility. Further, most of the measures are difficult to quantify and are dynamic in the sense that their values would tend to change with time as various system

parameters (e.g. demand, product design, technology etc.) are expected to change with time. In view of these, it can be concluded that the aforesaid problems of decision making would be complex even if few of these flexibility measures are taken into consideration.

The design decisions in FMS include selection of machines, tools, jigs, fixtures, materials handling system, part and tool storage system, loading/unloading system related to manufacturing; and selection of computer hardware and software, design of databases, information system and decision support system for decision making. The operational decisions which tend to be highly dynamic in FMS,include aggregate planning, part mix selection, allocation of resources, operations assignment, scheduling, monitoring and control.

Although the above mentioned decisions are also shared by conventional manufacturing systems, the complexity is much higher in case of FMS owing to the objective of maximization of resource utilization and of emulating a flow line production process in dynamic environment of multi-product and medium-variety production system with multifunctional and programmable machines and facilities.

Several approaches have over time evolved to cope up with the above mentioned complexities in decision making (Ham et al. (1985)). Some of these are mentioned below.

— Some of the basic principles of Industrial Engineering, viz. specialization, simplification, standardization of parts, materials and process routes.

— Part oriented production approach based on the production of

Maximum number of machines in a group $= \sum_{m=1}^{M} |N_m|$.

The values given above for the various factors are considered to be the default values in use of the heuristic.

HEURISTIC

Phase I

Step 0: Let P be the set of parts and M be the set of machine types. Find for each part $n \in P$, the part requirement given as:

$$PR_n = \sum_{m \in C(n)} a_{nm}.$$

Find also for each machine type $m \in M$, requirement on machine type MR_m and admissibility factor AF_m given as:

$$MR_m = \sum_{n \in \underline{C}(m)} a_{nm}$$

and $$AF_m = \frac{MR_m}{|N_m|}.$$

Now sequence all the parts in descending order of the number of machine type used.

Step 1: The first part in the sequence is chosen to form the first group. The machine cell will have all the machine types that are used by the part. The part family will have this part and all those parts whose more than half requirement can be met by the machine cell formed presently.

Step 2: Compute the rejectability factor for each machine type assigned to the machine cell as the ratio of sum of requirements of unassigned parts on the machine type to

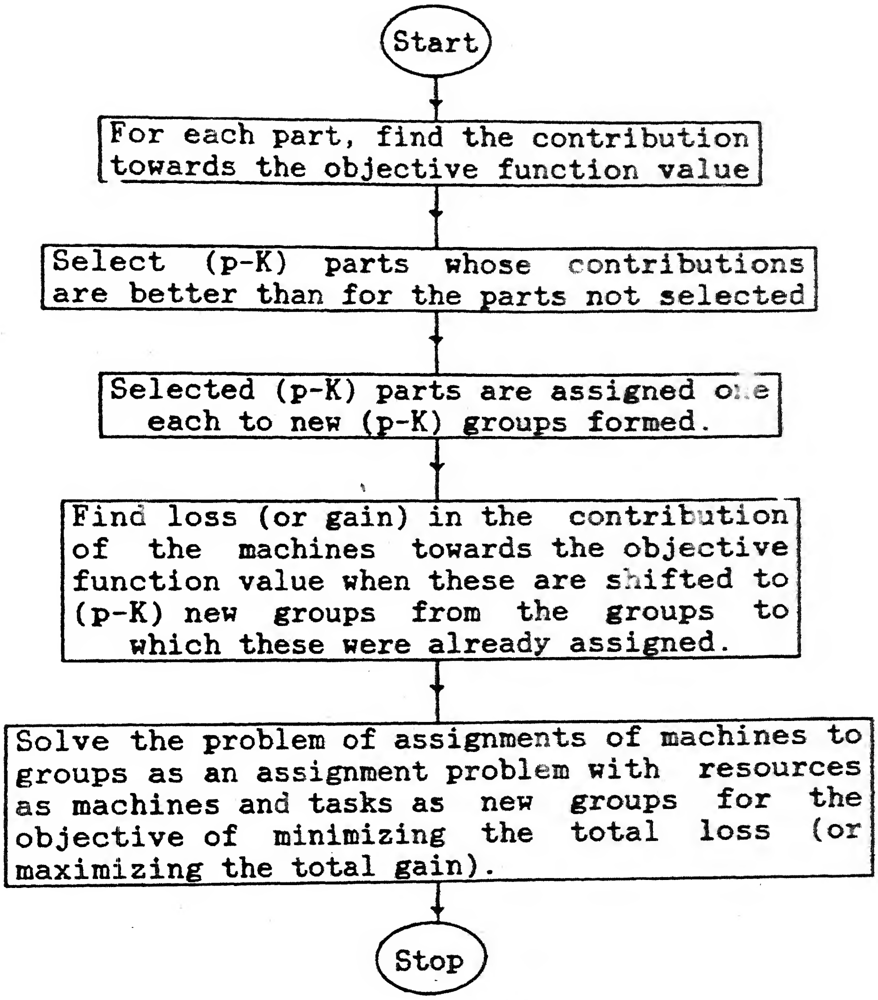


Figure 5.12: Flowchart Showing Details for Creating the Required Number of Additional Groups in Simple Grouping Case.

common components of various products in larger batches requiring the special setups only for special parts.

- Material Requirements Planning (MRP), a production planning and control system that, based on a master production schedule of end products, calculates the requirements of components at different levels of bills of materials of each product and determines the exact times of need for these components by using the information from item master data file and inventory file.
- Group Technology (GT), a concept to increase the production efficiency of the system by grouping various parts and products with similar design and/or production process.

Concurrent to the development and application of the above major popular approaches, several optimization procedures have also evolved to solve the design and operational level decision problems. The development in computers and computing technology have enormously helped in solving these complex optimization problems. Computers have also became inevitable in storage, retrieval and processing of huge information required for decision making in multi-product environment. The concepts of database management, decision support system, artificial intelligence and expert system are also being increasingly applied to analyze and solve various problems in manufacturing.

The present work focuses on the applications of GT on certain aspects of design and operational problems of FMS. A concise summary of fundamental concepts of GT appears in the subsequent sections followed by the scope and organization of the present work.

1.1 GROUP TECHNOLOGY: AN OVERVIEW

Group Technology (GT) is a philosophy applied for organizing, planning and control of industrial systems. An industrial system, in general, is viewed as a set of interrelated entities with interdependent attributes to achieve a common goal of producing useful products and/or services. The approaches for GT application are normally based on exploitation of similarity or the commonness amongst the entities to form *groups* or *clusters*. Traditionally, GT has been closely associated with manufacturing systems in which raw materials are transformed into the desired shape and size with the help of machines, tools and other related facilities. The major entities in manufacturing for the purpose of GT application are parts and machines. The groups of parts and machines are called as part families and machine cells, respectively.

The underlying philosophy behind GT is sufficiently general and is equally applicable for clustering in other diversified areas, such as, (i) Behavioral and social sciences (psychology, etc.), (ii) Earth Sciences (geography, etc.), (iii) Engineering Sciences (pattern recognition, artificial intelligence, etc.), (iv) Information and Policy Sciences (Information retrieval, etc.) (vi) Medicine and (v) Life Sciences. A detailed list of some of these can be had from Anderberg (1973).

The philosophy behind GT application in manufacturing can be best described by the definition of clustering given by Anderberg (1973).

"The objective is to group either the data units or the variables into clusters such that the elements within a cluster

constraints on machine capacity, again a different solution is obtained as can be seen from Tables 5.13 and 5.14.

Table 5.11: Operation Assignment Details for the Case When Machine Capacity Constraint is not Included and the Restriction on Group Disjointedness is Relaxed.

Objective function value = 94

Part	Operation	Assigned Machine
1	1	1
	2	-
	3	3
2	4	2
	5	6
3	6	1
	7	3
4	8	4
	9	5
	10	6
5	11	4
	12	5
6	13	4
	14	5

Table 5.12: Group Configuration Details for the Case When Machine Capacity Constraint is not Included and the Restrictions on Group Disjointedness is Relaxed.

Group #	Part-Family	Machine-Cell
1	{1, 3}	{1, 3}
2	{2, 4, 5, 6}	{2, 4, 5, 6}

unique routing is relaxed. The models proposed consider the same amount of tool slot consumption for an operation irrespective of the type of machine and the tool.

Identifying or developing suitable methodology for obtaining solutions to the 0-1 nonlinear integer programming problems appears to be an important issue. Attempts have been made to linearize the nonlinear terms and then to solve the problem using some standard integer programming approach. Stecke and Talbot (1983) reported that the use of these models (Stecke, 1983) in finding the solutions of real life problems would require enormous computational effort and substantial computer time, and thus the models may not be practical for real time applications. They proposed two algorithms for minimizing the part movement and another one for balancing (or unbalancing) the loads on machines of equal (unequal) size.

Shanker and Tzen (1985) proposed mathematical models for a random FMS. Machine loading and tool allocation decisions are arrived at while deciding in an integrated manner the part mix to be manufactured during the planning period. The constraints considered are the same as included by Stecke (1983) except for the constraint on the fixed routing. Besides this, parts are considered explicitly for assignment. The problem is formulated as a nonlinear mixed integer programming problem for two different objectives: (i) minimization of the total of unbalance in the workloads of machines, and (ii) minimization of the total of unbalance in the workloads of machines and the total number of late jobs. They also proposed heuristics for these two situations. A simulation model is presented to evaluate the system

have a high degree of natural association among themselves while the clusters are relatively distinct from one another".

A similar definition of GT given by Groovers (1987) is :

"Group Technology is an approach that seeks to identify those attributes of a population that permit its members to be called into groups, sometimes called families. The members of each particular group possess attributes that are similar."

In case of manufacturing where the elements (members) to be grouped are parts and machines, the association is generally based on design and manufacturing attributes of the parts. For this reason, Gallagher and Knight (1986) state the concept behind GT as:

"The basic concept is relatively simple: identify and bring together related or similar parts and processes to take advantages of the similarities which exist, during all stages of design and manufacture."

Traditionally, the concepts of GT in manufacturing have been used for: (i) standardization of design of parts, (ii) standardization of processing and routing and (iii) setup of work cells (Remmerwaal et al (1983)). The production system incorporating the GT concepts for layout of equipment and facilities is called as cellular manufacturing system. The increased adoption of GT philosophy in production is mainly attributed to the advantages listed below (Ham et al (1985), Gallagher and Knight (1986), Burbidge (1988)).

(i) Reduction in Setup

This is one of the major advantages of GT applications. Reduction in setup activities and times results by identifying

(v) Each tool type is limited in number.

The points mentioned above show that the production system is reasonably flexible and the parts can be manufactured in alternative ways. The loading models developed consider such generalities of the system and are formulated using the notations given below. These notations also assist in further understanding of the manufacturing scenario.

Subscripts

n : part

i : operation

j : machine

k : tool

Parameters

N = total number of parts

M = total number of machines

T = total number of tools

a_n = production volume of part n

$B(n)$ = set of the operations of part n

b_{ni} = total number of occurrences of operation i on each unit of part n

C_j = total capacity of machine j

U_j = capacity of tool magazine on machine j

W_j = a weight associated with machine j showing its relative significance

s_k = slot requirement of tool type k

V_k = number of copies available of tool type k

$t(i)$ = set of tools capable of performing operation i

c_{nijk} = cost of processing a single occurrence of operation i of part n using machine j and tool type k

groups of parts that have common requirements such as of tools, jigs, fixtures and machines. Producing the similar parts together will naturally require lesser changeover of these. By using nicely designed group tools, jigs and fixtures, setup times can further be reduced to a large extent.

(ii) Reduction in Materials Handling

By having group layout, i.e. by dedicating the required machines to each group of similar parts, significant reduction in materials handling and its related cost is achieved.

(iii) Reduction in Lead Time

As a consequence of lesser setup time requirement, the lead time reduces.

(iv) Increase in Throughput Rate

Because of the shortened setup time and reduced materials handling, the parts require much lesser time for their complete processing, and as a result of this throughput rate increases considerably.

(v) Increased Utilization of Resources

A heavily loaded traffic system normally faces the problem of traffic congestion and because of which machines, sometimes, starve waiting for the jobs to be loaded next on them. Traffic congestion may also lead to the blocking of machines where machines have to wait for the jobs to be unloaded from them. Blocking may also occur because of full in-process storage space that need to be cleared by materials handling system. All these factors result into under utilization of machines and other manufacturing resources. However, because of lesser materials handling in cellular manufacturing, these problems do not occur

requirement values. In fact, the slot requirement for a tool is counted as the number of slots physically covered by it, and the number is increased by one if the distance between the shank periphery and the extreme most slot covered by the tool is greater than or equal to the half of the distance between two consecutive slots (d_s). The tools as I and II normally lead to the problem of compatibility. Such tools will not require additional slot if on its both the sides some compatible tools can be placed. Otherwise, additional slots will be required which at the most could be equal to two. Thus, the scheme can be claimed to project average requirements of the tools.

It is felt that the above proposed method of computing the slot requirement would, to a great extent, account for slot saving due to overlap. Therefore, in view of this, the slot savings due to overlap need not be included in formulation. It is this consideration which has been used in the formulation developed in this chapter and the savings due to overlap is not considered explicitly.

In view of the discussions made in the previous paragraphs, the constraint that the total slot requirement of the tools to be assigned to a machine should not exceed the capacity of the magazine, can be written as:

$$\sum_{k=1}^{T} s_k \, Z_{jk} \leq U_j \qquad j = 1, \ldots, M. \tag{6.4}$$

In case when slot requirement of a tool is dependent on the machine on the tool magazine of which it is going to be placed,

frequently and thus resource utilization is comparatively high. Increased machine utilization is also achieved because of the reduction in setup activities.

(vi) Reduction in In-Process Inventory

As a result of reduced throughput time parts spend comparatively less time in the system, and thus their average in-process inventory level falls.

(vii) Better Space Utilization

As mentioned earlier, the flow of materials in group layout is close to that in product-layout. Further, since the in-process inventory level is low, not many inter-process stores will be required and the space required for each store may be comparatively less. This not only results into better utilization of space, but also cuts other tangible and intangible costs such as for rent, electricity, maintenance, etc..

(viii) Better Production Planning and Control

As compared to the problem of production planning and control in process layout manufacturing system where all the parts and all the resources (man, machine, material, etc.) are considered simultaneously for decision making, the problem in cellular manufacturing system is comparatively much simplified because of the involvement of lesser number of parts and machines. This cuts the size and the complexity of the problem to be analyzed for production planning, and a better plan is obtained requiring lesser efforts and time. Further, supervision of small shops leads to a better control, and not many progress chasers are required.

(xi) <u>Better Customer Service</u>

A better production plan along with better control mechanism normally leads to a more reliable production. This along with increased throughput rate, outputs of better quality, reduced manufacturing cost is expected to generate a fast response, early delivery and much better customer service.

(ix) <u>Better Quality</u>

As compared to the variety of products and operations handled by a worker in functional layout production system, the variety in group-layout is less. This leads to a faster learning and increasing specialty in carrying out certain smaller number of operations. Because of these factors, drastic reduction is achieved in number of rejects, and at the same time increase in the level of quality is achieved.

(x) <u>Reduced Cost of Production</u>

As mentioned before, the application of GT concepts in manufacturing is aimed at achieving benefits of mass production such as of lower direct and indirect labor cost. This in cellular manufacturing is achieved because of the cut in various kind of wastage such as in setting up activities and rejects, increased productivity of labour, requirement of less skilled workers, etc..

(xii) <u>Reduction and Rationalization of Product Variety</u>

While grouping the similar parts together, one is in a position to identify the functional characteristics of the products which make them different from the others. If the differences are insignificant and not essential to be maintained, then these can be removed leading to reduction in product variety and thus to a better cost-efficient production.

not exceed the available capacity, can be written as:

$$\sum_{n=1}^{N} \sum_{i \in B(n) \cap \underline{B}(j)} \sum_{k \in t(i) \cap \underline{t}(j)} t_{nijk}\, x_{nijk} \le C_j \qquad j = 1, \ldots, M. \tag{6.14}$$

(ii) Unique routing and tooling

This constraint is expressed as:

$$\sum_{j \in m(i)} \sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} = 1 \qquad n = 1, \ldots, B; \quad \forall\, i \in B(n). \tag{6.15}$$

(iii) Slots used by the tools allocated to a machine

The constraint that the slot requirement of the tools that can be loaded on a machine should not exceed the capacity of the tool magazine on it, can be written as:

$$\sum_{k = \underline{t}(j)} s_k\, Z_{jk} \le U_j \qquad j = 1, \ldots, M. \tag{6.16}$$

(iv) Copies of each tool type

This constraint can be written as:

$$\sum_{j \in \underline{m}(k)} Z_{jk} \le V_k \qquad k = 1, \ldots, T. \tag{6.17}$$

(v) Tool allocation for the assigned operation

The constraint is written as:

$$Z_{jk} - x_{nijk} \ge 0 \qquad n = 1, \ldots, N; \ \forall\, i \in B(n); \quad j \in m(i);\ k \in t(i) \cap \underline{t}(j). \tag{6.18}$$

DECISION VARIABLES

$$x_{nijk} = 0 \text{ or } 1 \qquad n = 1, \ldots, N; \ \forall\, i \in B(n) \quad j \in m(i) \text{ and } \forall\, k \in t(i) \cap \underline{t}(j) \tag{6.19}$$

Let Q_i be the number satisfying the relation

$$\sum_{p=1}^{Q_i-1} 2^{p-1} < a_n . b_{ni} \le \sum_{p=1}^{Q_i} 2^{p-1}.$$

Then, the variable X_{nijk} can be replaced by

$$\sum_{p=1}^{Q_i} 2^{p-1} . \delta_{ijkp}, \text{ where } \delta_{ijkp} \in \{0,1\}.$$

For example, if for an operation i of part n, $a_n = 10$ and $b_{n,i} = 3$, then $Q_i = 5$. If the total number of assignments of operation i to the combination of tool k and machine j is 11, then $\delta_{ijk1} = 1$, $\delta_{ijk2} = 1$, $\delta_{ijk3} = 0$, $\delta_{ijk4} = 1$ and $\delta_{ijk5} = 0$.

For the objective of minimizing the maximum workload on machines, the formulation of the problem will be as given below.

MODEL (M6.6)

Minimize L

Subject to:

$$\sum_{n=1} \sum_{i \in B(n) \cap \underline{B}(j)} \sum_{k \in t(i) \cap \underline{t}(j)} T_{nijk}\, x_{nijk} \le L \quad j=1, \ldots, M \qquad (6.27)$$

and (6.16), (6.17), (6.24), (6.25), (6.26) and (6.20).

EXAMPLE

Consider the example (Example 6.3) for which the data are given in Tables 6.15, 6.16 and 6.17. The solution for objective of minimizing the total processing cost is given in Tables 6.18 and 6.19, and that for the objective of minimizing the maximum workload on machines in Tables 6.20 and 6.21.

(xiii) Progress Towards Automation

Reduced product variety and grouping of similar parts together increase the possibility of automating the production system which may further economize the production process. Further, while designing a new product, the help of existing drawings and design procedures of similar other parts can be taken. This process can easily be automated and implemented on computers.

(xiv) Simplified Estimating and Work Measurement

Because of the grouping of similar parts together, the estimation for resource requirements and preparation of master production schedule becomes somewhat rational and practical, and can be done with much ease.

(xv) Simplified Accounting System

In conventional batch manufacturing system, cost accounting is a difficult exercise because of the large number of products sharing common facilities. However, in cellular manufacturing systems, it becomes simple because of the lesser number of products involved.

(xvi) Improved Job Satisfaction, Morale and Communication

As mentioned earlier, in cellular manufacturing a worker has to perform comparatively lesser number of operation types on small range of similar part types. This creates a feeling of being less loaded and thus provides better job satisfaction. Besides this, working on a product that is to be completely processed by a group of workers forms the basis for having pride in its accomplishment and thus results in a high morale. Further, in cellular manufacturing the labours and supervisors are better organized having

Table 6.22: Operation Assignment Detail for the Objective of Maximizing the Part Processing Flexibility (Example 6.3).

Part	Operation	Machine	Tool Type
1	1	1	1
		3	2
	2	4	2
	3	2	3
2	4	2	3
		5	3
	5	2	3
		3	5
3	6	5	3
	7	4	2
	8	3	2
	9	3	5
4	10	5	3
	11	1	1
		2	1
5	12	2	3
	13	3	5
	14	3	2

Minimum part processing flexibility = 3 units

better communication links, and because of this workers participation increases, and improvement of job and implementation of quality circles becomes possible. Fazakerley (1976) had shown by carrying out a survey that in the case of the cellular manufacture, the labours are more satisfied and frustration level is low. Knight (1989) has shown that the cellular manufacture provides a good platform for the improvement of communication process, and he proposes, in addition, a structure that using computers can be utilized for the betterment of communication.

In a recent survey carried out by Wemmerlov and Hyer (1989(a), 1989(b)), it has been mentioned that implementation of GT philosophy in manufacturing is not a new phenomenon in developed countries and is taking place at a much faster pace at present because of the changing market conditions and consumer behaviour. Their surveys clearly exhibit achievements of the advantages of GT application mentioned above. The problems of cellular manufacturing reported by them, such as, out-of-cell operations, volume/capacity imbalance, load imbalance, and the problems faced during machine breakdown can be reduced by properly designing the cells. It may require a thoughtful layout of cells and then machines in the cells.

1.2 GT AND FMS

As mentioned earlier, an FMS is highly capital intensive and is required for those manufacturing scenarios where product variety and product volume are small to medium. Therefore, care must be taken for utilizing resources in FMS optimally, allowing for practically negligible idle times. A requirement of this kind

Chandrasekharan, M.P. and Rajagopalan, R. (1987), "ZODIAC - An Algorithm for Concurrent Formation of Part-Families and Machine Cells," International Journal of Production Research," Vol. 25, No. 6, 835-850.

Chandrasekharan, M.P. and Rajagopalan, R. (1989), "Groupability: An Analysis of the Properties of Binary Data Matrices for Group Technology," International Journal of Production Research, Vol. 27, No. 6, 1035-1052.

Chen, R. (1983), "Solution of Minisum and Minimax Location - Allocation Problems with Euclidean Distances," Naval Research Logistics Quarterly, Vol. 30, 449-459.

Choobineh, F. (1988), "A Framework for the Design of Cellular Manufacturing Systems," International Journal of Production Research, Vol. 26, No. 7, 1161-1172.

Christofides, N. and Viola, P. (1971), "The Optimum Location of Multi-centres of a graph," Operational Research Quarterly, Vol. 22, 145-154.

Co, H.C. and Araar, A. (1988), "Configuring Cellular Manufacturing Systems," International Journal of Production Systems, Vol. 26, No. 9, 1511-1522.

Dao, T., and Boisclair, J. (1989), "Microcomputer Interactive Software for Optimization of (GT) Machine-Component Group Formation," Computers and Industrial Engineering, Vol. 17, Nos. 1-4, 624-629.

Darrow, W.P. and Gupta, J.N.D. (1989), "Integrating GT and MRP Systems Through Lot Sizing and Scheduling," Computers and Industrial Engineering, Vol. 16, No. 2, 287-296.

Das, C., Patel, D.K. and Rath, S.K. (1989), "New Efficient Cluster Identification Algorithms," International Journal of Management and Systems, Vol 5, No. 3, 173-184.

Datta, S.P., Lashkari, R.S., Nadoli, G. and Ravi, T. (1986), "A Heuristic Procedure for Determining Manufacturing Families from Design-Based Grouping for Flexible Manufacturing Systems," Computers and Industrial Engineering, Vol. 10, No. 3, 193-201.

Deo, N. "Graph Theory: with Applications to Engineering and Computer Science," Prentice Hall Inc., Englewood Cliffs, N.J., USA, 1989.

El-Essawy, I.F.K. and Torrance, J. (1972), "Component Flow Analysis - An Effective Approach to Production Systems Design," Production Engineer, Vol. 51, No. 5, 165-170.

Fazakerley, G.M. (1976), "A Research Report on the Human Aspects

will ask for more serious implementation of GT while considering the added flexibility of flexible machines and material handling system (MHS) in FMS. Though the manufacturing philosophy of FMS and conventional manufacturing system are entirely at variance, yet the basic philosophy behind GT application remains the same: exploit the commonness.

In the recent years, researchers have started advocating that for the application of GT in FMS, it is rational to give more emphasis on common resources consumed rather than on geometrical similarity for group formation purposes (Tuffentsammer and Arndt (1983), Kusiak (1985)). Deviations in part-geometry do not play all that important role as they do in conventional manufacturing system (Warnecke et al (1986)).

FMS supports (i) automatic MHS which is capable of transferring workpieces between any set of machines automatically, and (ii) automatic loading and unloading of palletized parts on workcenters and MHS. Machines have generally more than one tool on their tool magazines and any of the tool can be picked up almost instantaneously using automatic tool-changer. Therefore, to a large extent parts can be processed in FMS in a cost-efficient way. This is in contradiction with situations in conventional manufacturing systems where to achieve benefits of GT it is necessary to process a batch of a part completely before taking another batch of a different part on any workstation.

In conventional manufacturing system, GT is normally used to find systematic group layouts. Obviously, grouping of machines will be physical; whereas in FMS environment, GT concepts are and should be used to find logical grouping besides determining

Zelenovic, D.M. and Tesic Z.M. (1988), "Periodic Batch Control and Group Technology," International Journal of Production Research, Vol. 26, No. 3, 539-552.

physical ones. These logical groupings help in simplifying planning, loading and scheduling problems in FMS by decomposing a big size problem into subproblems of much smaller size. The grouping depends upon the part-mix to be manufactured. Therefore, groups of parts and machines remain dynamic in nature and would change as and when status of any of the part and machine changes.

An implementation of GT would consider many factors such as type and age of production equipment, flexibility required; and the success in doing so will depend largely upon the understanding of GT concepts and level of manufacturing expertise (Tuffentsammer and Arndt (1983)).

1.3 METHODOLOGIES FOR GT CELL FORMATION

Over a period since inception of GT philosophy, various kinds of approaches have been proposed for solving GT problems. Some of the solution procedures are evaluative while others are generative in nature. Some of the methods solve problem of part family and machine cell formation simultaneously while others do it sequentially. Various approaches and methodologies are discussed in the following subsections. Surveys of these methods have been made available by several researchers from time to time. A recent survey on some of the GT techniques and approaches is given by Chu (1989).

1.3.1 Production and Component Flow Analysis

Production Flow Analysis (PFA) is an approach for forming groups of parts and machines and had been proposed by Burbidge (1963, 1971, 1973). This was the first analytical approach for GT application. In this analysis, first *factory-flow-analysis* is done and then *department-flow-analysis* is carried out. Later is

further subdivided into *Group-Analysis* and *Line-Analysis*. The approach was further clarified by solving some real-life problems. In this case, groups of parts and machines are decided simultaneously.

The other less popular technique proposed by El-Essawy and Torrence (1972) is based on Component Flow Analysis (CFA). CFA was claimed to be different from PFA in the way the groups are formed. However, many authors report not to find any substantial difference between the two techniques. CFA categorizes the components based on their manufacturing requirements and then tries to determine the machine groups.

1.3.2 Coding and Classification Systems

Another popular scheme of GT applications is based on coding of parts and classification procedures. The codes are based on part design attributes or part manufacturing attributes, or mix of the two attributes. Some of the popular codes have been compiled and listed by Ham et al. (1985). Codes based on design attributes generally consider geometrical shape of the parts and its dimensions, and thus are useful for promoting design standardization. Codes based on manufacturing attributes are quite useful for computer aided process planning, tool design and other production oriented function, and thus suitable for GT cell formation. Codes using both types of attributes provide naturally the advantages of the both. Presently more emphasis is being given to the classification using feature based geometric models because of their amenability for GT applications (Shah and Bhatnagar (1989)) .

The codes and classification schemes have been used for the

purpose of cell formation by many researchers. Mills (1986) discusses the use of classification schemes for the design of a cellular manufacturing facility. A number of methods that use codes have been described by Kusiak (1983, 1984). Some of these methods are:

(i) nearest neighbour method,

(ii) matrix formulation,

(iii) integer programming formulation

and (iv) X-neighbourhood clustering.

The part grouping problem formulated as an integer programming problem requires number of part families to be formed as specified.

The coding systems are quite useful for standardization of part design process and make the task simpler. These also serve the purpose of designing manufacturing cells nicely, but mostly for conventional manufacturing systems because majority of the codes were developed keeping conventional tools in mind giving more importance to absolute size and rough shapes of the parts. However, in FMS where machines are mostly flexible, versatile and multi-functional, the use of these conventional codes may not be appropriate. Similar observations and comments have been made by Tuffentsammer and Arndt (1983), and also by Warnecke et al (1986). In addition to the part geometry, one should consider clamping devices, jigs and fixtures to be used.

1.3.3 Expert Systems

Other important approach for group formation uses the concepts of artificial intelligence. Expert system for GT cell formation can incorporate several quantitative and qualitative

factors which otherwise are hard to be included in decision making.

Kusiak (1987(a)) has proposed architectures for expert system for GT application. The expert systems can be used in a stand alone mode using simple rules or in a tandem mode along with some algorithms for solving the problem. The details of some of the algorithms have been presented in his other paper (1987(b)). In another paper, Kusiak (1988) considers more detailed problem considering various relevant constraints such as on capacity of machines, type and capacity of materials handling system, etc..

1.3.4 Graph Theoretic Approaches

Rajagopalan and Batra (1975) used graph-partitioning approach for machine cell formation. First, a graph is defined with machines as vertices. The edges between various nodes represent the association between machines obtained using Jaccard's similarity coefficient. After this, the graph is partitioned to obtain machine cells and then components are assigned to these cells.

Kumar et al. (1986) consider for grouping purpose a machine-part incidence matrix which may have numbers even other than binary ones. The problem is formulated as a K-decomposition problem considering limit on cell size. The cell size is defined as the sum of the number of parts and machines assigned to that cell. A heuristic approach is suggested for solving the grouping problem. Similar formulation and heuristic approach is suggested for grouping of parts and fixtures by these authors (1985).

Vanneli and Kumar (1986) discuss the method of finding minimal bottleneck machines (parts) which when duplicated

(subcontracted) will result perfect part-machine grouping. For solving this problem, a heuristic approach has been suggested. In the other paper (1987), they consider similar problem with the objective of minimal number or minimal cost of parts which when subcontracted will lead to disaggregated manufacturing. The constraints on the maximum number of machines in each cell are also considered for the purpose of determining the number of groups. For this problem also, similar heuristic approach has been used.

In another approach for group formation, various cost elements were considered by Askin and Chiu (1987). Mathematical formulation includes the constraints on the number of workers and machines in each group. However, the heuristic solution methodology that uses graph partitioning approach is proposed for a much simplified problem.

Chandrashekharan and Rajagopalan (1986) used the K-means method for evolving the groups of parts and machines. First, the grouping problem is posed as an exercise of decomposing a bipartite graph to get a limit on the maximum number of groups that can be formed. Next, some arbitrary seeds are used to find groups of parts and machines. Through diagonalization, part families are assigned to machines-cells taking help of group-efficiency measures. Further, ideal seeds are used to remove unnatural groups obtained at previous stage. In the other paper (1987), they proposed that initial grouping can be obtained by a combination of arbitrary and representative seeds or artificial and representative seeds, or also by using natural seeds. It was further suggested that when initial groups are identified using natural seeds, one can find related part families and machine

cells without using ideal seeds at a subsequent stage after the process of diagonalization is over.

1.3.5 Optimization Techniques

Mazzola et al (1987) viewed the grouping problem as a problem of grouping the machines. All the machines in a group are to be of the same type and are to be identically configured. The machine grouping problem along with loading problem is considered in the framework of material requirements planning and is formulated as a 0-1 mixed integer programming problem.

Ventura et al (1987) came with the equivalent formulation as was proposed by Kumar et al (1986) for the problem of grouping of parts and machines. However, a different approach is adopted for solving the problem. Lagrangian relaxation technique is used in conjunction with subgradient optimization technique for solving the problem.

Co and Arrar (1988) solved the problem of grouping of parts and machines in three stages. In the first stage, operations of the parts that can be performed by the machines of a particular type are assigned to such machines considering the objective of minimizing the maximum deviation between assigned workload and available capacity on the machines. Thereafter, at second stage, extended form of King's algorithm (1980) is used for arranging machines according to similarities between various operations. The third stage is a direct search algorithm which determines size and composition of cells using the results of the previous stage. Kumar et al (1987) use minmax approach for solving combined problem of grouping and loading formulated as a multistage multiobjective optimization model. Each of the objectives of

grouping and loading enjoy the same weightage.

The integer programming formulation of the grouping problem (Kusiak (1983)) was suitably modified by Kusiak (1987) for the grouping situations where it is possible to process a job using different combination of machines. Though the number of groups needs to be specified, no limitation is imposed on the number of parts or machines that can be assigned to a group. The grouping procedure aims at determination of part families knowing which machine cells are to be determined later. This problem was also formulated as a generalized assignment problem by Shtub (1989). An important feature of these models has been the consideration given to flexibility in carrying out operations which traditionally were to be performed each on a single machine.

The grouping problem formulated as a 0-1 integer programming problem by Gunasingh and Lashkari (1989a) tries to group the machines taking as input some already decided part families, and considering the constraints on the number of machines of each type and limitations on the total number of machines in a group. In their other article (1989b), a hierarchical method is proposed which first determines machine cells based on similarity in part-processing. Next, the parts are allocated to appropriate machine groups based on their processing requirements. The two problems are formulated as 0-1 integer programming problems.

1.3.6 Clustering

A rich literature is available on the techniques to be used for clustering. Depending upon the basic mechanism for group formation, the techniques can be classified as given in the following subsections.

1.3.6.1 SIMILARITY COEFFICIENT BASED APPROACHES

Probably the oldest method that is based on the use of similarity coefficient method is by McAuley (1972). He uses Jaccard's similarity coefficient for the purpose of group determination. The procedure was termed as *single linkage clustering methodology*. The method has been improved by Seifoddini (1989a) and termed as *average linkage clustering methodology*. The improved method results in a better machine grouping having lesser number of exceptional elements (Seifoddini (1989b)). In the single linkage clustering methodology, the groups with two machines (one machine from each of the two groups) are combined without accounting for the similarity or dissimilarity between other pairs of machines. However, in the average clustering methodology, groups are combined on the basis of maximum average similarity between machines.

The definition of Jaccard's similarity coefficient was enriched by Seifoddini (1988, 1989(c)) by including certain additional part attributes such as required volumes and required movements between a pair of machines for a part. Such considerations suitably weigh the parts and the resultant grouping is based on more practical measure of intercell movements.

Waghodekar and Sahu (1984) developed a heuristic procedure for machine cell formation. The method, called as MACE, determines the groups of machines using similarity of product type coefficient. They also proposed a different similarity coefficient which is based on the total flow of common parts between a pair of machines. Further, suggestions are also given for handling exceptional elements. The procedure simply finds machine

cells and using this information one has to find part families later.

A different approach has been taken by Dutta et al (1986) where design based grouping of parts is converted into manufacturing based grouping of parts. For this purpose, a measure for finding dissimilarity between two parts based on their tooling requirements is suggested. The final grouping, of course, corresponds to minimal overall dissimilarity.

Another different measure for similarity determination was proposed by Choobineh (1988). He considers the similarity between the sequences of the operations of two parts of a pair while calculating the similarity coefficient. He proposed a two phase procedure for group formation. The first stage based on the similarity measures finds part families, and the second stage determines machine cells and the number of machines of various types to be assigned to each of these cells.

Similarity as a cell bond strength was defined by Studel and Ballakur (1987). It is determined using the processing time requirements of parts on machines. The groups of machines are determined in two stages. In the first stage, a chain of machines that maximizes the sum of bonds between pair of consecutive machines of the chains is obtained, and then machine cells are determined using any of the two approaches described in the paper while considering cell size restrictions.

Linear cell clustering algorithm proposed by Wei and Kern (1989) also uses similarity coefficient for group formation. The similarity coefficient proposed can be considered to be a modified version of the coefficient used by Kusiak (1987) which is defined

for a pair of parts (or process plans) instead of for a pair of machines. The requirements of both the machines of a pair by a part is weighed more compared to when both of them are not required by these parts.

A weighing mechanism different from that proposed by Seifoddini (1989(c)) mentioned and discussed earlier had been developed by Mosier and Taube (1985). The structure of the two proposed similarity coefficients are also different from that of Seifoddini (1989(c)). Mosier (1989) further analyzed these two similarity measures, called as average similarity coefficient and multiplicative similarity coefficient, along with some other similarity coefficients for different grouping methodologies. Out of these coefficients, Jaccard's similarity coefficient and modified multiplicative weighted similarity coefficient were reported to generally provide comparatively better results.

A different methodology has been proposed by Gunasingh and Lashkari (1989(a), 1989(b)) where similarity is defined based on common tools.

As can be seen from the above, most of the similarity coefficients are calculated based on any of the following parameters or their combinations:

(a) machine requirements

(b) tooling requirements

(c) sequence among the operations of the parts

(d) production volumes of the parts.

In a recent analysis by Taboun (1989), it has been shown that groups obtained based on the above factors differ from each other significantly. Based on the statistical comparison of the resul-

ts, they find those grouping strategies to be the best that use similarity coefficient based on the required machines.

1.3.6.2 ARRAY BASED APPROACHES

The clustering of parts and machines based on ordering of rows and columns was initially proposed by King (1980). The method called as Rank Order Clustering (ROC) was intended to give block-diagonal structure by ordering rows and columns of part-machine matrix considering row and column vectors as binary numbers. With increasing size of the matrix, binary number values corresponding to these vectors increase and this limits the size of the matrix which could be handled by the method when implemented on computer. To overcome this problem, a modified version of this clustering technique, named as ROC2, was proposed ky King and Nakornchai (1982). But this too could not remove the weakness of ROC of its being biased towards initial disposition of 0-1 numbers in part-machine matrix because of which the method leads to poor block-diagonal structure even in case of well structured matrix. An effort was made by Chandrashekharan and Rajagopalan (1986) to remove the above mentioned shortcoming of ROC in their proposed method MODROC, being an extension and improved version of ROC. In MODROC, ROC procedure is used in conjunction with block and slice method that results into sets of intersecting machine cells and nonintersecting part families. Then, a hierarchical clustering method is used for obtaining final machine and part groups using the associations between various pairs of machine-groups and some preassigned threshold value.

A different way of overcoming the problem of dependency of ROC on initial arrangement of numbers in the matrix was suggested

ts, they find those grouping strategies to be the best that use similarity coefficient based on the required machines.

1.3.6.2 ARRAY BASED APPROACHES

The clustering of parts and machines based on ordering of rows and columns was initially proposed by King (1980). The method called as Rank Order Clustering (ROC) was intended to give block-diagonal structure by ordering rows and columns of part-machine matrix considering row and column vectors as binary numbers. With increasing size of the matrix, binary number values corresponding to these vectors increase and this limits the size of the matrix which could be handled by the method when implemented on computer. To overcome this problem, a modified version of this clustering technique, named as ROC2, was proposed ky King and Nakornchai (1982). But this too could not remove the weakness of ROC of its being biased towards initial disposition of 0-1 numbers in part-machine matrix because of which the method leads to poor block-diagonal structure even in case of well structured matrix. An effort was made by Chandrashekharan and Rajagopalan (1986) to remove the above mentioned shortcoming of ROC in their proposed method MODROC, being an extension and improved version of ROC. In MODROC, ROC procedure is used in conjunction with block and slice method that results into sets of intersecting machine cells and nonintersecting part families. Then, a hierarchical clustering method is used for obtaining final machine and part groups using the associations between various pairs of machine-groups and some preassigned threshold value.

A different way of overcoming the problem of dependency of ROC on initial arrangement of numbers in the matrix was suggested

ts, they find those grouping strategies to be the best that use similarity coefficient based on the required machines.

1.3.6.2 ARRAY BASED APPROACHES

The clustering of parts and machines based on ordering of rows and columns was initially proposed by King (1980). The method called as Rank Order Clustering (ROC) was intended to give block-diagonal structure by ordering rows and columns of part-machine matrix considering row and column vectors as binary numbers. With increasing size of the matrix, binary number values corresponding to these vectors increase and this limits the size of the matrix which could be handled by the method when implemented on computer. To overcome this problem, a modified version of this clustering technique, named as ROC2, was proposed ky King and Nakornchai (1982). But this too could not remove the weakness of ROC of its being biased towards initial disposition of 0-1 numbers in part-machine matrix because of which the method leads to poor block-diagonal structure even in case of well structured matrix. An effort was made by Chandrashekharan and Rajagopalan (1986) to remove the above mentioned shortcoming of ROC in their proposed method MODROC, being an extension and improved version of ROC. In MODROC, ROC procedure is used in conjunction with block and slice method that results into sets of intersecting machine cells and nonintersecting part families. Then, a hierarchical clustering method is used for obtaining final machine and part groups using the associations between various pairs of machine-groups and some preassigned threshold value.

A different way of overcoming the problem of dependency of ROC on initial arrangement of numbers in the matrix was suggested

by Shanker and Zhu (1987) where ROC is used on initialized matrix. In case where clear block-diagonal structure does not emerge, a manual modification procedure is used for obtaining final part families and machine cells. The method proposed by Das et al (1989) also requires manual adjustment and identification of groups when perfect grouping is not obtained using ROC. This method, however, does not initialize the part-machine matrix and thus may not be superior to the methodology proposed by Shanker and Zhu (1987). Simultaneous grouping methodology proposed by Roy and Sengupta (1989) also tries to initialize the matrix indirectly while sorting the machines and components based on their requirements. In their approach, groups of parts and machines are formed after sorting based on the similarity computed for various parts and machines.

Direct Clustering Algorithm proposed by Chan and Milner (1983) also first tries to initialize the binary matrix for removing the dependency of the ROC algorithm on the configuration of the initial matrix. Except this feature of the algorithm, it is reported by Khator and Irani (1987) to be a poor version of the ROC algorithm.

1.3.6.3 OTHER HEURISTICS

The within-cell utilization based clustering heuristic WUBC, proposed by Ballakur and Studel (1987) tries to form part families and machine cells considering processing time requirements of the parts. A part is assigned to a cell where its maximum number of operations can be completed, and a machine is assigned to a cell only when the workload factor of this machine for that cell is greater than or equal to some preassigned cell-admissibility

factor. The methodology incorporates constraints on the limit of cell size and considers multiple copies available of the machines of the various types.

The occupancy value method proposed by Khator and Irani (1987) uses a different procedure for developing block-diagonal structure. The diagonal structure instead of being developed as in ROC, gets developed progressively starting from the northwest corner of the matrix.

Srinivasan et al (1990) use assignment technique for the purpose of group formation. The assignment matrix is a similarity coefficient matrix, the elements of which represent similarity between various pair of machines. Closed loops in the form of subtours represent initial machine cells. From these machine cells, the final groups of parts and machines are determined.

A heuristic proposed by Harhalakis et al (1990) attempts to determine the machine cells between which the intercell traffic is minimum. The grouping is determined in two phases. In the first phase, machines are assigned to cells based on the volume (or cost) of the material flow, and in the other phase this solution is improved, if possible. The intercell traffic is computed considering the sequence between the operations.

The heuristic approach proposed by Al-Qattan (1990) starts forming the group by selecting a seed machine. A seed machine is the one that is required by the minimal number of parts. Grouping is obtained by branching from this seed machine. A branch is bounded on completed part. The heuristic may provide a number of alternative solutions based on the presence of a seed part, and thus increase the flexibility for manufacturing cell designer by

having a good number of options to be evaluated for such purposes.

Askin and Subramanian (1986) instead determine the configuration of groups based on economic considerations such as costs on inventory carrying, materials handling, etc.. The method suggested uses a three phase procedure. In the first phase, part and machine groups are determined using a procedure similar to that of King and Nakornchai (1982). In the second phase, The groups thus formed are combined if the economic benefits are resulted. In the third phase, these groups are further condensed subject to constraints on machine capacity.

A successful application of fuzzy mathematics for part family formation has been made by Xu and Wang (1989). Two different approaches introduced for family formation are fuzzy classification and fuzzy equivalence. The computer programme developed has certain other features such as control on the group formation process. The groups are formed by specifying either the threshold similarity coefficient value or the number of groups to be formed. Further, the features of the parts based on which the similarity coefficients are computed, can directly be described from the engineering drawings.

1.4 GT AND OTHER FUNCTIONS

Various problems of production planning and control such as grouping, loading, scheduling, sequencing, inventory planning and control, capacity planning, etc. are highly interrelated although operationally different from each other. These problems when solved independently and/or hierarchically may result the solutions which could be locally optimal or, sometimes, may contradict each other. For example, the batch size of the parts from the

viewpoint of inventory management may be different from that determined using batching algorithms used for loading and scheduling. Thus, to get the maximum advantages the problems should be considered in an integrated manner. Such an effort will involve many parameters in decision making and consequently the size of the problems may be large. In some cases, the modeling of the integrated problems may be a difficult task. Because of this reasons, such an approach may be prohibitive. Nevertheless, the integration and decision making for few of the functions combined together may not be very difficult and the benefits of the results obtained from such an analysis may outweigh the difficulties faced in such an analysis. Moreover, because of the development of the high speed computers it has been possible to obtain solutions within reasonable amount of time, and due to this the computational problem related with integration does not remain to be very serious. A detailed review of several manufacturing planning and control functions in the context of GT applications can be had from Ham et al (1985).

In the following subsections, some efforts which have been made in past for integration of GT with other manufacturing functions are presented.

1.4.1 Material Requirements Planning

As mentioned earlier, material requirement planning (MRP) is a popular technique for planning the requirements of the raw materials and the components that are required for manufacturing the end product. In addition, batch size of the products and the periods in which these will be produced are also being specified. Group technology tries to identify the similar parts that may be

procured at the same time resulting more economic lot sizes and lesser requirement of setup times.

Sato et al (1978) described a methodology that can be used for integrating GT and MRP, and thus for obtaining a much better schedule for manufacturing of parts. A similar conceptual framework has been used by Darrow and Gupta (1989). However, the procedure followed remains somewhat different. The parts with similar manufacturing requirements are grouped together. The elements of the part family are combined into a single batch for processing. Reduction in setup cost is achieved by minimizing the sequence dependent setup times between components in the part family. A case study presented shows substantial reduction in setup and carrying costs obtained using GT with MRP as compared to when all the parts are considered independently as in the traditional MRP.

1.4.2 Loading

Loading is an operational problem where decisions are to be taken mainly for the assignments of operations to the various machines. These decisions normally depend upon the configuration of the groups and their constituent machines. Groups designed considering simply parts and their machine requirements may not be practical from the loading point of view. The groups determined in the usual way applying the concepts of GT may turn out to be disjoint, but in the presence of considerations of machine capacity and assigned workloads on them, the groups may not remain disjoint. Because of this fact, it may not be possible to achieve the benefits of GT as conceived. Therefore, efforts are made to find decisions for grouping and operation assignment problems

considering them together in an integrated manner. Recently, some efforts have been made by various researchers to find grouping and loading decisions simultaneously considering both the problems combined together.

Kumar et al (1987), as mentioned earlier, formulate grouping and loading problems as a multiobjective and multistage optimization problem. The groups of machines are allowed to have different types of machines. This is in contradiction with the approach taken by Mazzola et al (1987) where machine cells are to have machines of the same type that are identically configured.

1.4.3 Scheduling and Sequencing

As mentioned earlier, GT aims at reducing the overall setup activities. This reduction can only be achieved if the batches of the different parts are sequenced in a way that requires minimum changeover of setups needed to process them. A good sequence should have in it the parts with similar manufacturing requirements together. The similar parts must further be sequenced properly so as to minimize the sequence dependent setup times and costs. For this purpose, the parts should be grouped based on the required manufacturing resources such as jigs, fixtures, part-grippers, etc..

For the reasons and advantages listed in the preceding paragraphs, it was advocated by many to use GT approach for part sequencing and scheduling. A good number of algorithms have also been proposed. For example, computer software MISCHEDULE developed by Remmerwaal et al (1983) tries to improve the sequence of the jobs to be processed by a facility. Jobs are sequenced taking into account their geometry, dimensions and other manufacturing

requirements. The software has been developed keeping traditional manufacturing system in mind. Recently, an effort has been made by Hitomi et al (1989) to develop a scheduling algorithm for an FMS kind of environment. The models proposed take into consideration automation of MHS and also for setup. Algorithm is also developed for determining the optimal number of buffer spaces.

Since the mathematical models or the algorithms proposed may not consider many practical constraints and several parameters which may affect or govern the scheduling decisions, they suffer from certain limitations. These can be overcomed by using simulation technique. Miles and Batra (1986), and Abdin and Mohamed (1988) use this technique and also show the way as how effectively simulation can be used for determining or analyzing a scheduling and sequencing policy.

1.5 RESEARCH ISSUES

As mentioned in the previous section, GT needs to be integrated with other manufacturing functions. The procedures and the concepts used for such integration must properly consider the manufacturing system, its attributes and characteristics, and also the constituent equipment. For conventional manufacturing, the geometry and size of parts may play an important role but may not be adequate for FMS. In FMS where the machines are multifunctional, programmable and capable to accommodate a range of combinations of system parameters, the grouping decisions should normally consider the comprehensive manufacturing resource requirements of the parts. Efforts and investigations are required to see as how these parameters can suitably be considered for group formation. Further, during integration of GT with other manufact-

uring functions their relevance has to be analyzed. The advantages achieved from integration and the computational requirements and the size of the problems are also required to be properly evaluated. It may require a thorough understanding of the system to suggest the ways for finding solutions to various operations management problems as which are to be determined in an integrated manner and the other that are to be determined hierarchically.

1.6 PRESENT WORK

Some of the important details of the problems considered for the present study, and also the details of the organization of the work carried out in this thesis are delineated below.

1.6.1 Scope

The purpose of the present thesis is to study and solve the problem of grouping combined with loading problem in an FMS environment. The approach of combined analysis is expected to provide a better result in terms of the quality of the groups and the operationally feasible solution for the operations management problems.

The flexible manufacturing system considered for the present study is assumed to be a general one where the parts can be manufactured in a number of ways using different combinations of machine and tool types. In other words, a part can have alternative process plans. The alternative process plans for a part emerge because of the machine flexibility, process flexibility, routing (or scheduling) flexibility, and sometimes due to operations flexibility.

Grouping of parts with alternative process plans, known as *generalized grouping,* though is more complex than for the case of

simple grouping where a part has only one process plan, but offers solutions to design and certain operational problems in FMS. One of the operational problem is the loading problem which seeks to assign the part operations to machines. The results of the present study can also be effectively used for the traditional manufacturing systems involving simple grouping.

While addressing the problem of generalized grouping, it is felt that the existing measures of similarity are inadequate for grouping purpose in generalized situations. Thus, a set of new similarity coefficients are introduced to find similarity between a pair of process plans of the parts based on the individual or combined requirements of manufacturing resources, such as, machines, tools, etc.. The new coefficient is christened as *requirement compatibility* coefficient.

For the generalized grouping problems, efforts have been made for developing mathematical models. A variety of approaches are suggested for the determination of groups of parts and machines. An analysis is also carried out to check and suggest the suitability and relevance of these approaches for different grouping scenarios. The models proposed incorporate several other practical considerations such as the size of part families and machine cells, capacity of machines and number of copies of each machine type.

1.6.2 Organization

In Chapter II, the new similarity coefficient measures -absolute and relative requirement compatibility- are introduced for finding similarity between two process plans of the parts. A comparison between these proposed measures and some other popular

similarity measures are made for their discriminating characteristics. In Chapter III, mathematical models for the generalized grouping are developed following p-median problem formulation approach, and the associated solution methodologies are discussed. In Chapter IV, p-centre problem formulation approach for the generalized grouping problem has been discussed. The formulations of the grouping problems developed in the Chapters III and IV use the requirement compatibility coefficients introduced in the Chapter II. In Chapter V, a different approach is discussed where groups are determined using the concepts of graph partitioning. For some of these models, heuristic procedures are also suggested.

One of the important features of the models developed in the Chapters III, IV and V is solutions for operations management problems that also emerges while deciding the groups. Thus, these models can be seen to simultaneously provide results for loading problems also. Some of the models, however, do not provide a complete solution to operations management problem which can separately be obtained using the loading models developed in Chapter VI. These loading models can also be used for finding decisions about tool allocation while maximizing the flexibility in carrying out the operations.

Numerical examples are presented for the clarity of the model formulation and solution methodologies, and also for a better understanding of the models and the related grouping scenarios. At the end of each chapter summary of its contents are presented along with some conclusions drawn from the analysis carried out in the respective chapters. For a better and overall comprehension of the complete work, the summary is also presented in Chapter VII.

In addition, the scope of the work is discussed and suggestions are made for further improvements, implementation and integration.

CHAPTER II

Requirement Compatibility and Other Measures of Commonality for GT Applications : Relationships and Discriminating Characteristics

2.1 INTRODUCTION

As emphasized in the Chapter I, Group Technology (GT) is one of the most rational approaches for increasing productivity in medium-volume medium-variety production environment especially for automated discrete part manufacturing. The advantages of GT for production can be achieved by forming groups of parts and machines. The groups are determined exploiting commonality between parts and/or machines, and thus, discriminating similar parts and machines from dissimilar ones. The commonality between two parts is found generally by identifying machines required (or not required) by both the parts, and is measured on the basis of either the number of such machines or the processing time requirement on these machines.

The commonality, for the purpose of grouping, has been defined in several ways by various researchers. Majority of the measures define the commonality between a pair of parts or a pair of machines. Recently, some measures have been proposed to describe the commonality between a pair of a part and a machine/machine group.

<u>Commonality Between a Pair of Parts (or a Pair of Machines)</u>

Some of the measures for commonality are: Jaccard's similarity coefficient (JSC) used by Rajagopalan and Batra (1975), product type similarity coefficient (PSC) defined by Wagohdekar

and Sahu (1984), cell bond strength (CBS) proposed by Ballakur and Studel (1987) and simple matching similarity coefficient (MSC) discussed by Chandrashekharan and Rajagopalan (1989). These measures originally proposed for finding commonality between a pair of machines can also be used for a pair of parts.

The JSC is defined as the ratio of the number of common machines to the total number of the machines required by the two parts, whereas the MSC as the ratio of the sum of the number of machines required and not required by both the parts of a pair to the total number of machines considered for grouping. The normalized city block distance, as discussed by Chandrashekharan and Rajagopalan (1989), is complement of MSC and, therefore, can be found by subtracting the value of MSC from one. The PSC is defined as the product of two ratios each obtained after dividing the number of machines commonly required by both the parts by the number of total machines required by the corresponding parts. The CBS, based on the processing time of the parts, is defined as the sum of two ratios each found after dividing the sum of the processing requirements on common machines by the total requirement of their corresponding part. The MSC and the normalized city block distance, as described by Kusiak (1984), can also be calculated based on Minkowski and Hamming metrics, respectively.

The measures mentioned in the preceding paragraph have the following limitations in their use:

(i) Parts should have only one process plan, i.e., alternative process plans are not admissible.

(ii) The usual incidence matrix showing the requirement of machines for the various parts cannot consider the number of times a machine is required for processing the various operations of a part. In other words, the multiple operations of a part requiring the same machine are not differentiated.

(iii) The sequence in which the operations of a part are to be performed on the various machines, can also not be considered.

Subsequent efforts by some researchers have tried to remove some of these limitations. Choobineh (1988) modified the definition of the Jaccard's similarity coefficient and extended it to include the sequence among the various operations for finding the similarity between the parts. Kusiak (1987) defined another commonality measure for finding the commonality between two process plans. This measure, primarily proposed for use in the situations where parts have alternative process plans, is an extension and modification of MSC. The measure proposed by Seifoddini (1988) is again an extension of JSC which incorporates the volume of the part as well as the sequence of the operations. However, the method of including the effect of the sequence of the operations is different from that of Choobineh (1988).

The measures mentioned above are symmetric in nature, i.e. the commonality of a part (or a process plan) with respect to some other is the same as the commonality of the latter with respect to the first. Moreover, these measures fail to recognize the requirements of a part on machines that are also needed by the other part. In other words, these measures do not find the compatibi-

lity of a part with respect to another. Further, instead of finding compatibility, these commonality measures find similarity, so to say closeness between the two parts based on their requirements on machines. This may cause an uncomfortable situation when a part is found to be almost compatible with the other part, and the other part being less compatible with respect to the first one. In such cases, the first part happens to be a strong candidate for grouping with the other part, but the low ratings for the closeness between the two parts obtained from the use of these measures may suggest the two parts to be put into two different groups.

<u>Commonality Between a Pair of a Part and a Machine/Machine Group</u>

Two measures which find commonality of a part with respect to a machine/machine-group have been proposed by Gunasingh and Lashkari (1989a, 1989b) and have been termed as compatibility. The compatibility of a part with respect to a machine group has been defined as the ratio of the twice of the number of common tools between the parts and machine group to the sum of the numbers of different types of tool required by the parts and that available in the machine group (Gunasingh and Lashkari(1989b)). The other measure describing compatibility of a part with respect to a machine has been defined as a proportion of the total requirement of tools or processing times of the part which can be met by the machine (Gunasingh and Lashkari (1989b)).

Some of the observations regarding the limitations of the commonality measures made for the case of a pair of parts (or a pair of machines) are also applicable for the above measures of commonality defined for a pair of a part and a machine/machine-group.

Suitability of Commonality Measures for Grouping

As described by King and Nakornchai (1982), the commonality measures based on the similarity coefficients are aggregate measures and their manipulation causes information loss; nevertheless, these are widely used because of their simplicity in use. However, the power of discriminating similar parts and similar machines from dissimilar ones depends upon the commonality measure chosen for the group formation. A comparison of the discriminating power of similarity measures by Chandrashekharan and Rajagopalan (1989) shows that the Jaccard's similarity coefficient has better discriminating characteristic than the normalized city block distance and the simple matching similarity coefficient. However, Studel and Ballakur (1987) claim the cell bond strength to be a better measurement than the Jaccard's similarity coefficient.

At present, no suitable measure of commonality seems to exist for GT applications for generalized situations which do not suffer from the previously mentioned limitations of being unable to accommodate the alternative process plans of the various parts, and to differentiate the operations requiring the machines of the same type and capability. In such generalized situations which can be encountered with most production situations for discrete part manufacturing, the grouping exercise needs to consider a very general framework that is expected to turn the related problems quite interesting and complex.

Present Work

In the present chapter, attempts are made to define for the above generalized case, a measure of commonality, to be called as

Requirement Compatibility. The compatibility of a process plan with respect to another can be determined on the basis of either the number of common machines and/or tools, the sum of the processing time of its operations requiring these machines, or the number of such operations. Its two propositions, called Absolute Requirement Compatibility (ARC) and Relative Requirement Compatibility (RRC), measure the compatibility of a process plan with respect to another absolutely or relative to its total requirement, respectively. These are suitably restated for the situations where parts have no alternative process plans. Further, the concepts of compatibility are extended for a pair of a process plan and a machine. The relationship of RRC for pair of parts are established with the other commonality measures CBS, JSC and PSC. It is shown that the CBS, JSC and PSC can be derived from the RRC. The mathematical relationships established are used to analyze the discriminating power of all these measures. Illustrations presented are found to support the results of the above analysis, and are also used to compare the discriminating power of the simple matching similarity coefficient. A simulation study is performed which not only determines the relative discriminating power of these measures, but also gives insight into the numerical values of the measures and provides guidelines for the selection of threshold values of commonality measures for the purpose of grouping.

2.2 REQUIREMENT COMPATIBILITY

In this section, the requirement compatibility is first defined in various alternative ways and then their expressions are

developed. The grouping environment considered contains parts with the following characteristics.

(a) A part may have alternative process plans.

(b) Operations in a process plan though requiring machines of the same type and capability may be considered distinct because of the different requirements of setups, jigs, fixtures, tools or cutting parameters.

(c) An operation for its processing on different types of machines may require different combinations of pallets, jigs, fixtures, tools, etc.

It can be noted from the classification of the operations that the production environment considered here is more general as compared to that reported in the previous works. Recently, Gunasingh and Lashkari (1989b) have envisaged a production situation where operations are classified based on the tool requirements and each operations is to be performed using only one tool type on the machines on which the required tool type is available. However, the situation might be somewhat different in practice. For example, a slot on a prismatic part can be made using either a slotter or a milling machine. But the use of the slotter machine will require a slot cutter, whereas the use of milling machine a milling cutter.

Thus, it can be seen that the production environment considered in the present work is quite general accommodating various aspects of grouping and considering flexibility for carrying out the operations.

Some Definitions

For further discussions, following terminologies are used:

Common Machines for two process plans are those machines which are required for processing some operations of both the process plans.

Compatible Operations are those operations of a pair of process plans which require the common machines and/or common tools for their processing. Whereas, for a pair of a process plan and a machine, the compatible operations are those which can be performed on the machine forming the pair or require the tools that can be loaded on the machines.

Matched Processing Requirement of a process plan P_I with respect to process plan P_{II} is the sum of processing times of those operations of P_I which require machines and/or tools that are common to process plan P_{II}. Whereas, matched processing requirement of a process plan P with respect to a machine M is the sum of processing times of those operations that can be performed on the machine M or require tools that can be loaded on the machine M.

Total Requirement (R) of a process plan is taken as the total number of operations, the total number of machines or tool types required for processing all the operations, or the sum of processing times of the operations in the plan.

Matched Requirement (RM) of a process plan P_I with respect to process plan P_{II} is taken as the number of the common machines and tools, the number of the compatible operations of the process plan P_I with respect to P_{II}, or the matched processing requirement of the process plan P_I with respect to P_{II}. Whereas, matched

requirement of a process plan P with respect to a machine M is either the number of compatible operations of the plan P or the matched processing requirement of the plan P with respect to the machine M.

Absolute Requirement Compatibility (ARC) of a process plan P_I with respect to process plan P_{II} is the matched requirement of P_I with respect to P_{II}; whereas the absolute requirement compatibility of a process plan P with respect to a machine M is the matched requirement of the plan P with respect to the machine M.

Relative Requirement Compatibility (RRC) of a process plan P_I with respect to process plan P_{II} is the ratio of the matched requirement of P_I with respect to P_{II} to the total requirement of P_I; whereas the relative requirement compatibility of a process plan P with respect to a machine M is the ratio of the matched requirement of the plan P with respect to the machine M to the total requirement of the process plan P.

Notations

Following notations will be used for further deliberations:

M = the total number of machine types

T = the total number of tool types

F_i = set of alternative process plans for part i; ($|F_i| \geq 1$)

G_m = set of tools which can be loaded on machine type m (m = 1, ..., M)

w_i = production volume of part i

P_n^i = n^{th} process plan of part i

$N(P_n^i)$ = the total number of operations in process plan P_n^i

$R(P_n^i)$ = the total requirement of process plan P_n^i

$B(k:P_n^i)$ = machine type required for processing k^{th} operation in process plan P_n^i

$C(k:P_n^i)$ = tool type required for processing k^{th} operation in process plan P_n^i

$D(P_n^i)$ = set of machine types required for processing all the operations in process plan P_n^i

$$= \bigcup_{k=1,..,N(P_n^i)} B(k:P_n^i)$$

$E(P_n^i)$ = set of tool types required for processing all the operations in process plan P_n^i

$$= \bigcup_{k=1,..,N(P_n^i)} C(k:P_n^i)$$

$T(k:P_n^i)$ = processing time requirement of k^{th} operation in process plan P_n^i

$RM(P_u^i,P_v^j)$ = matched requirement of process plan P_u^i with respect to plan P_v^j

$RM(P_u^i,m)$ = matched requirement of process plan P_u^i with respect to machine type m

$$\delta(c,d) = \begin{cases} 1 & \text{if } c = d \\ 0 & \text{otherwise} \end{cases}$$

$$\mu(c) = \begin{cases} 1 & \text{if } c \geq 1 \\ 0 & \text{otherwise} \end{cases}$$

$$X(k:P_u^i,P_v^j) = \begin{cases} 1 & \text{if the machine type required for processing } k^{th} \text{ operation in process plan } P_u^i \text{ is also needed for processing at least one operation in plan } P_v^j \\ 0 & \text{otherwise} \end{cases}$$

$$= \mu \left[\sum_{\hat{k}=1}^{N(P_v^j)} \delta\left\{B(k:P_u^i),\ B(\hat{k}:P_v^j)\right\} \right]$$

$$\hat{X}(k:P_u^i,P_v^j) = \begin{cases} 1 & \text{if the tool type required for processing } k^{th} \text{ operation in process plan } P_u^i \text{ is also needed for processing at least one operation in plan } P_v^j \\ 0 & \text{otherwise} \end{cases}$$

$$= \mu \left[\sum_{\hat{k}=1}^{N(P_v^j)} \delta\left\{C(k:P_u^i),\ C(\hat{k}:P_v^j)\right\} \right]$$

$$Y(k:P_u^i,P_v^j) = \begin{cases} 1 & \text{if the tool and machine types required for processing } K^{th} \text{ operation in } P_u^i \text{ are together needed for processing at least one of the operations in plan } P_v^j \\ 0 & \text{otherwise} \end{cases}$$

$$= \mu \left[\sum_{\hat{k}=1}^{N(P_v^j)} \delta\left\{B(k:P_u^i),B(\hat{k}:P_v^j)\right\} .\delta\left\{C(k:P_u^i),C(\hat{k}:P_v^j)\right\} \right]$$

$$\hat{Y}(k:P_u^i,P_v^j) = \begin{cases} 1 & \text{if the tool and machine types required for processing } k^{th} \text{ operation in process plan } P_u^i \text{ are needed for processing some operations in plan } P_v^j \\ 0 & \text{otherwise} \end{cases}$$

$$= X(k:P_u^i,P_v^j) \ . \ \hat{X}(k:P_u^i,P_v^j)$$

$$Z(P_u^i,m) = \begin{cases} 1 & \text{if machine type m is required for processing some operation in process plan } P_u^i \\ 0 & \text{otherwise} \end{cases}$$

$$= \mu \left[\sum_{k=1}^{N(P_u^i)} \delta\left\{m, B(k:P_u^i)\right\} \right]$$

$$\hat{Z}(P_u^i,t) = \begin{cases} 1 & \text{if tool type t is required for processing some operation in process plan } P_u^i \\ 0 & \text{otherwise} \end{cases}$$

$$= \mu \left[\sum_{k=1}^{N(P_u^i)} \delta\left\{t, C(k:P_u^i)\right\} \right]$$

$CBS_{i,j}$ = cell bond strength between the parts i and j

$JSC_{i,j}$ = Jaccard's similarity coefficient for the parts i and j

$MSC_{i,j}$ = simple matching similarity coefficient for the parts i and j

$PSC_{i,j}$ = product type similarity coefficient for the parts i and j

$ARC(P_u^i,P_v^j)$ = absolute requirement compatibility of process plan P_u^i with respect to plan P_v^j

$ARC(P_u^i,m)$ = absolute requirement compatibility of process plan P_u^i with respect to machine type m

$RRC(P_u^i,P_v^j)$ = relative requirement compatibility of process plan P_u^i with respect to plan P_v^j

$RRC(P_u^i,m)$ = relative requirement compatibility of process plan P_u^i with respect to machine type m

2.2.1 For a Pair of Process Plans

Using the definitions and the notations given above, the absolute requirement compatibility and the relative requirement

compatibility can be expressed as:

$$ARC(P_u^i, P_v^j) = RM(P_u^i, P_v^j) \qquad \forall\, i;\ \forall\, j;\ u = 1, \ldots, |F_i|;\ v = 1, \ldots, |F_j| \qquad (2.1)$$

$$RRC(P_u^i, P_v^j) = \frac{RM(P_u^i, P_v^j)}{R(P_u^i)} \qquad \forall\, i;\ \forall\, j;\ u = 1, \ldots, |F_i|;\ v = 1, \ldots, |F_j| \qquad (2.2)$$

where $RM(P_u^i, P_v^j)$ and $R(P_u^i)$ can be computed using bases of the processing times, the number of common machines and tools, and the number of the operations. Since the requirements of a process plan can have complete to no matching with the requirements of another plan, the bound on the values of matched requirement can be given as:

$$0 \le RM(P_u^i, P_v^j) \le R(P_u^i) \qquad \forall\, i;\ \forall\, j;\ u = 1, \ldots, |F_i|;\ v = 1, \ldots, |F_j|.$$

From the equations (2.1) and (2.2), and the bound for the matched requirement given in the above expression, the following can easily be observed:

$$0 \le ARC(P_u^i, P_v^j) \le R(P_u^i) \qquad \forall\, i;\ \forall\, j;\ u = 1, \ldots, |F_i|;\ v = 1, \ldots, |F_j|,$$

$$0 \le RRC(P_u^i, P_v^j) \le 1 \qquad \forall\, i;\ \forall\, j;\ u = 1, \ldots, |F_i|;\ v = 1, \ldots, |F_j|.$$

Expressions for the total and the matched requirements for the different bases are given in the following subsections.

2.2.1.1 BASED ON THE PROCESSING TIMES

(a) Total Requirement

$$R(P_u^i) = w_i \left[\sum_{k=1}^{N(P_u^i)} T(k:P_u^i) \right] \qquad \forall\ i; \quad u = 1, \ldots, |F_i| \tag{2.3}$$

(b) Matched Requirement

(i) On common machines

$$RM(P_u^i, P_v^j) = w_i \left[\sum_{k=1}^{N(P_u^i)} T(k:P_u^i)\ X(k:P_u^i, P_v^j) \right] \qquad \forall\ i;\ \forall\ j; \quad u = 1, \ldots, |F_i|; \quad v = 1, \ldots, |F_j| \tag{2.4}$$

(ii) On common tools

$$RM(P_u^i, P_v^j) = w_i \left[\sum_{k=1}^{N(P_u^i)} T(k:P_u^i)\ \hat{X}(k:P_u^i, P_v^j) \right] \qquad \forall\ i;\ \forall\ j; \quad u = 1, \ldots, |F_i|; \quad v = 1, \ldots, |F_j| \tag{2.5}$$

(iii) On common machines and common tools

$$RM(P_u^i, P_v^j) = w_i \left[\sum_{k=1}^{N(P_u^i)} T(k:P_u^i)\ Y(k:P_u^i, P_v^j) \right] \qquad \forall\ i;\ \forall\ j; \quad u = 1, \ldots, |F_i|; \quad v = 1, \ldots, |F_j| \tag{2.6}$$

2.2.1.2 BASED ON THE NUMBER OF OPERATIONS

(a) Total Requirement

$$R(P_u^i) = w_i\ N(P_u^i) \qquad \forall\ i; \quad u = 1, \ldots, |F_i| \tag{2.7}$$

(b) Matched Requirement

(i) On common machines

$$RM(P_u^i, P_v^j) = w_i \left[\sum_{k=1}^{N(P_u^i)} X(k:P_u^i, P_v^j) \right]$$

$$\forall\ i;\ \forall\ j;\quad u = 1, \ldots, |F_i|;\quad v = 1, \ldots, |F_j| \qquad (2.8)$$

(ii) On common tools

$$RM(P_u^i, P_v^j) = w_i \left[\sum_{k=1}^{N(P_u^i)} \hat{X}(k:P_u^i, P_v^j) \right]$$

$$\forall\ i;\ \forall\ j;\quad u = 1, \ldots, |F_i|;\quad v = 1, \ldots, |F_j| \qquad (2.9)$$

(iii) On common machines and common tools

$$RM(P_u^i, P_v^j) = w_i \left[\sum_{k=1}^{N(P_u^i)} Y(k:P_u^i, P_v^j) \right]$$

$$\forall\ i;\ \forall\ j;\quad u = 1, \ldots, |F_i|;\quad v = 1, \ldots, |F_j| \qquad (2.10)$$

2.2.1.3 BASED ON THE NUMBER OF MACHINES

(a) Total Requirement

$$R(P_u^i) = |D(P_u^i)| \qquad \forall\ i;\quad u = 1, \ldots, |F_i| \qquad (2.11)$$

(b) Matched Requirement

$$RM(P_u^i, P_v^j) = \sum_{m \in D(P_u^i)} Z(P_v^j, m)$$

$$\forall\ i;\ \forall\ j;\quad u = 1, \ldots, |F_i|;\quad v = 1, \ldots, |F_j| \qquad (2.12)$$

2.2.1.4 BASED ON THE NUMBER OF TOOLS

(a) Total Requirement

$$R(P_u^i) = |E(P_u^i)| \qquad \forall\ i;\ u = 1, \ldots, |F_i| \qquad (2.13)$$

(b) Matched Requirement

$$RM(P_u^i, P_v^j) = \sum_{t \in E(P_u^i)} \hat{Z}(P_v^j, t) \qquad \forall\ i;\ \forall\ j;\ u = 1, \ldots, |F_i|;\ v = 1, \ldots, |F_j| \qquad (2.14)$$

2.2.1.5 SOME COMMENTS AND OBSERVATIONS

Given below are the some observations and comments related to the definitions and the use of the expressions for the total and matched requirements described in the Sections 2.2.1.1 to 2.2.1.4.

(a) The expressions for the total and the matched requirements given in the Sections 2.2.1.1 and 2.2.1.2 are the same if the processing times of all the operations in the various plans of the parts are equal to one, i.e.

$$T(k:P_u^i) = 1 \qquad \forall\ i;\ k = 1, \ldots, N(P_u^i);\ u = 1, \ldots, |F_i|.$$

(b) When the operations are differentiated based on only the requirements of machine types, the expressions (2.7) and (2.11) for the total requirements developed using the respective bases of the number of operations and the number of machines, will be equivalent. The expression (2.12) for the matched requirement will, of course, be equivalent to the equation (2.8). Similarly, when the operations are differentiated on the basis of only required tool types, the equations (2.13) and (2.14) would become equivalent to the

equations (2.7) and (2.9), respectively. The equivalence between the various expressions will require the production volume w_i appearing in the equations (2.7), (2.8) and (2.9) to be equal to one.

The above observation will, in general, hold when the production volumes of all the parts are the same.

(c) From the discussions made above in (a) and (b), it can be seen that the expressions given in the Section 2.2.1.1 are quite general and can be used to find the total and the matched requirement for the other bases simply by making suitable classification of the operations and by making some suitable interpretations of production volumes of parts and processing times of the operations.

(d) The approach for finding requirements based on common tools or common machines can also be used to determine requirements based on other factors such as common jigs, common fixtures, etc.. Similarly, the approach for finding requirements based on the combination of common tools and machines can be extended for determining the requirements for other possible combinations of machines, tools, jigs, fixtures, etc.

(e) Grouping based on identification of common machines, pallets, tools, jigs, fixtures, machining parameters etc. or on any combination thereof is of importance in the manufacturing environment such as FMS comprising automated and multifunctional machines, automated MHS. To comprehend the relevance of the consideration of tools, etc. for grouping, let us consider two operations on two different parts which are similar to each other in terms of their tool, jig and machine

requirements. These two operations because of the different shape and size of the parts may require different kind of fixtures and thus may differ. In the other case, let us assume that the two operations of the same type are similar in terms of the requirements of jig and fixture but require different cutting parameters. If the cutting conditions of the two operations cannot be met by the same machine then the operations have to be loaded on two different machines of the same type but having different machining capabilities, and thus the operations will differ.

Further, the cost of pallets, fixtures or jigs for FMS are high and their use will depend upon the part, the machine or combination of the both, thus while forming groups of parts and machines due consideration should be given to required pallets, jigs, fixtures, etc.. The definitions of ARC and RRC can also be used for this purpose.

(f) For the case when the matched requirements are determined based either on the number of machine types (equation (2.12)) or on the number of tool types (equation (2.14)), the following will hold:

$$RM(P_u^i, P_v^j) = RM(P_v^j, P_u^i) \qquad \forall\ i;\ \forall\ j;\ u = 1, \ldots, |F_i|;\ v = 1, \ldots, |F_j| \qquad (2.15)$$

From the equations (2.1) and (2.15), it is obvious that the ARC computed on the bases of the number of machines or tools will be symmetric. That is,

$$\mathrm{ARC}(P_u^i, P_v^j) = \mathrm{ARC}(P_v^j, P_u^i) \qquad \forall\ i;\ \forall\ j;\ u = 1, \ldots, |F_i|\ \ v = 1, \ldots, |F_j|. \qquad (2.16)$$

(g) The equation (2.6) and (2.10) determine the matched requirement of process plan P_u^i with respect to plan P_v^j by summing the requirements of those operations in P_u^i for which the required combinations of tools and machines are also needed for processing some operations in P_v^j. In other words, the requirements of those operations in P_u^i are added which are similar to some other operations in P_v^j in terms of both machine and tool type requirements. In case when the matched requirement of P_u^i is based on those operations that require tools or machines common to some operations of plan P_v^j, the term $Y(k{:}P_u^i,P_v^j)$ should be replaced by $\hat{Y}(k{:}P_u^i,P_v^j)$ in the equations (2.6) and (2.10).

(h) ARC will be sensitive to the production volume of the parts, whereas RRC will not. In general, the number of intercell movements depends upon the production volume, the number of operations of each type and the availability of the tools and machines. Thus for the purpose of reducing intercell movements, groups should be determined using ARC computed on the basis of the number of operations and considering the production volume. ARC, however, as can be seen from its definition, does not consider the sequence among operations and thus can be considered to represent a optimistic savings in the number of intercell movements.

(i) The intercell movement cost, in general, will be different for the different parts owing to the raw material characteristics, size, shape and weight. The variable w_i can be

interpreted to incorporate the cost of movement where w_i is defined as the product of the production volume and the unit cost of movement for part i. The expressions for ARC, thus, would determine the commonality including the cost of movements.

2.2.2 For a Pair of Parts

In the case where the parts do not have alternative process plans, the formulations given in the Section 2.2.1 can be used for computing the requirement compatibility by substituting P_u^i and P_v^j by i and j, respectively, and for brevity placing them as subscripts, e.g. RM(i,j) will be denoted as RM_{ij}. The resulting expressions for the absolute requirement compatibility and the relative requirement compatibility, from the equations (2.1) and (2.2), are:

$$ARC_{i,j} = RM_{i,j} \qquad \forall\ i;\ \forall\ j, \qquad (2.17)$$

and

$$RRC_{i,j} = \frac{RM_{ij}}{R_i} \qquad \forall\ i;\ \forall\ j. \qquad (2.18)$$

The expressions showing bounds on the values of ARC and RRC are:

$$0 \le ARC_{i,j} \le R_i \qquad \forall\ i;\ \forall\ j,$$

and

$$0 \le RRC_{i,j} \le 1 \qquad \forall\ i;\ \forall\ j.$$

Given below are the resultant expressions for the total and the matched requirements determined based on the number of machines. These expressions will be used in Section 2.3 to establish relationship of RRC with other commonality measures.

(a) Total Requirement

$$R_i = |D_i| \qquad \forall\ i \qquad (2.19)$$

(b) Matched Requirement

$$RM_{i,j} = \sum_{m \in D_i} Z_{j,m} \qquad \forall\ i;\ \forall\ j. \qquad (2.20)$$

The expressions given in the equations (2.15) and (2.16), showing the symmetric property of the matched requirement and the absolute requirement compatibility, will simplify to:

$$RM_{i,j} = RM_{j,i} \qquad \forall\ i \text{ and } \forall\ j, \qquad (2.21)$$

and

$$ARC_{i,j} = ARC_{j,i} \qquad \forall\ i \text{ and } \forall\ j. \qquad (2.22)$$

2.2.3 For a Pair of a Process Plan and a Machine

The absolute requirement compatibility and the relative requirement compatibility as per their definitions can be expressed by equations (2.23) and (2.24) given below.

$$ARC(P_u^i, m) = RM(P_u^i, m) \qquad \forall\ i;\ u = 1, \ldots, |F_i|;\ m = 1, \ldots, M \qquad (2.23)$$

$$RRC(P_u^i, m) = \frac{RM(P_u^i, m)}{R(P_u^i)} \qquad \forall\ i;\ u = 1, \ldots, |F_i|;\ m = 1, \ldots, M \qquad (2.24)$$

The expressions for the total and the matched requirements for the different bases relevant to the pairs of parts and machines, are as follows.

2.2.3.1 BASED ON THE PROCESSING TIMES

(a) Total Requirement

It can be computed from the equation (2.3).

(b) Matched Requirement

The matched requirement of a process plan based on the processing times of the operations which are compatible to a machine, is given by the following expression:

$$RM(P_u^i, m) = w_i \left[\sum_{k=1}^{N(P_u^i)} T(k:P_u^i)\ \delta\left\{m,\ B(k:P_u^i)\right\} \right]$$

$$\forall\ i; \quad u = 1, \ldots, |F_i|; \quad m = 1, \ldots, M. \qquad (2.25)$$

2.2.3.2 BASED ON THE NUMBER OF OPERATIONS

(a) Total Requirement

The total requirement for this situation is computed from the equation (2.7).

(b) Matched Requirement

The matched requirement is expressed as:

$$RM(P_u^i, m) = w_i \left[\sum_{k=1}^{N(P_u^i)} \delta\left\{m,\ B(k:P_u^i)\right\} \right]$$

$$\forall\ i; \quad u = 1, \ldots, |F_i|; \quad m = 1, \ldots, M. \qquad (2.26)$$

2.2.3.3 BASED ON NUMBER OF TOOLS

(a) Total Requirement

The total requirement can be computed using the equation (2.13).

(b) Matched Requirement

It can be computed from the following equation.

$$RM(P_u^i, m) = \sum_{t \in G_m} \hat{Z}(P_u^i, t)$$

$$\forall\ i; \quad u = 1, \ldots, |F_i|; \quad m = 1, \ldots, M. \qquad (2.27)$$

2.2.3.4 SOME COMMENTS AND OBSERVATIONS

Similar to the observations made in the Section 2.2.1.5, given below are some observations and comments related to the definitions and equations (2.25), (2.26) and (2.27) of the matched requirement of a part computed with respect to a machine.

(a) The expressions for matched requirement based on processing time (equation (2.25)) and that based on the number of operations (equation (2.26)) are equivalent if the processing times of all the operations are the same and equal to one, i.e.

$$T(k{:}P_u^i) = 1 \qquad \forall\ i;\ k = 1, \ldots, N(P_u^i); \quad u = 1, \ldots, |F_i|.$$

(b) The matched requirement based on the number of operations (equation (2.26)) with $w_i = 1\ \forall\ i$ is equivalent to the matched requirement based on the number of tools (equation (2.27)) provided the operations are classified based on the requirement of tools.

(c) From the discussions made above in (a) and (b), it can be seen that the definition of the matched requirement based on the processing times (equation (2.25)) is the most general one.

(d) For the case when each of the parts has single process plan, the equations (2.25), (2.26) and (2.27) can be simplified by substituting P_u^i by i.

(e) The relative requirement compatibility defined on the basis of the number of tools for a pair of a part and a machine is similar to the compatibility index defined by Gunasingh and Lashkari (1989a) as the ratio of common tools between the part and the machine to the minimum number of tools available on the machine and the number required by the part.

(f) The definitions of requirement compatibility can be extended suitably for incorporation of the different combinations of pallets, tools, jigs, fixtures, etc. in a manner as discussed in the Section 2.2.1.5(e).

(g) ARC remains sensitive to the production volume of the parts, whereas RRC does not.

(h) The definitions of the matched requirement of a part with respect to a machine group can also be determined using the equation (2.27). However, G_m will denote the set of tools which can be loaded on the machines of the machine group.

2.3 RELATIONSHIP OF RELATIVE REQUIREMENT COMPATIBILITY WITH OTHER COMMONALTY MEASURES FOR A PAIR OF PARTS

In the following sections, the relationship between various measures of commonality are established and investigations and comments are made based on their interrelationship.

2.3.1 Mathematical Relationship

In this section, the mathematical relationship of the proposed relative requirement compatibility (RRC) with other commonality measures, viz., the cell bond strength (CBS), the

Jaccard's similarity coefficient (JSC), the product type similarity coefficient (PSC) and the simple matching similarity coefficient (MSC), is established. Though the definitions of CBS, JSC, PSC and MSC can be extended to find commonality between pair of process plans, for the simplicity these measures and RRC are considered for a pair of parts and are defined on the basis of the number of machines.

The expressions for MSC, CBS, JSC and PSC are given below:

$$MSC_{i,j} = \frac{M - R_i - R_j + 2\ RM_{i,j}}{M} \qquad \forall\ i;\ \forall\ j \qquad (2.28)$$

$$CBS_{i,j} = \frac{RM_{i,j}}{R_i} + \frac{RM_{j,i}}{R_j} \qquad \forall\ i;\ \forall\ j \qquad (2.29)$$

$$JSC_{i,j} = \frac{RM_{i,j}}{R_i + R_j - RM_{i,j}} \qquad \forall\ i;\ \forall\ j \qquad (2.30)$$

$$PSC_{i,j} = \frac{RM_{i,j}}{R_i} * \frac{RM_{j,i}}{R_j} \qquad \forall\ i;\ \forall\ j \qquad (2.31)$$

where the value for RM_{ij} is the same as given in the equation (2.20) and those of R_i and R_j are as given in the equation (2.19).

From the equation (2.18),

$$R_i = \frac{RM_{i,j}}{RRC_{i,j}} \qquad \forall\ i;\ \forall\ j, \qquad (2.32)$$

and

$$R_j = \frac{RM_{j,i}}{RRC_{j,i}} \qquad \forall\ i;\ \forall\ j. \qquad (2.33)$$

Using the symmetry property of $RM_{i,j}$ given in the equation (2.21), the equation (2.33) can be rewritten as:

$$R_j = \frac{RM_{j,i}}{RRC_{j,i}} \qquad \forall\ i;\ \forall\ j.$$

Using the equation given above and the equations (2.18) and (2.32), the equations (2.29), (2.30) and (2.31) can be written as:

$$CBS_{i,j} = RRC_{i,j} + RRC_{j,i} \qquad \forall\ i;\ \forall\ j, \qquad (2.34)$$

$$JSC_{i,j} = \frac{RRC_{i,j} \cdot RRC_{j,i}}{RRC_{i,j} + RRC_{j,i} - RRC_{i,j} \cdot RRC_{j,i}} \qquad \forall\ i;\ \forall\ j, \qquad (2.35)$$

$$PSC_{i,j} = RRC_{i,j} \cdot RRC_{j,i} \qquad \forall\ i;\ \forall\ j. \qquad (2.36)$$

From the equations (2.34) and (2.36), the equation (2.35) can be written as:

$$JSC_{i,j} = \frac{PSC_{i,j}}{(CBS_{i,j} - PSC_{i,j})} \qquad \forall\ i;\ \forall\ j. \qquad (2.37)$$

Since the value of RRC can at most be equal to one, following will be valid:

$$RRC_{i,j} + RRC_{j,i} - RRC_{i,j} \cdot RRC_{j,i} \leq 1 \qquad \forall\ i;\ \forall\ j.$$

Thus the equation (2.35), in view of the above inequality,

will transform to:

$$JSC_{i,j} \geq RRC_{i,j} \cdot RRC_{j,i} \qquad \forall\ i;\ \forall\ j.$$

The above relation in view of the equation (2.36) is:

$$JSC_{i,j} \geq PSC_{i,j} \qquad \forall\ i;\ \forall\ j. \qquad (2.38)$$

2.3.2 Observations on Interrelationship and Definitions of Commonality Measures

From the expressions of the definitions and the relations between RRC, CBS, JSC and MSC described and developed in the Sections 2.2.2 and 2.3.1, the following observations can be made:

(i) From the equations (2.34) and (2.36), it can be seen that the sum of the RRCs of two parts determined with respect to each other is equal to CBS between the two parts; whereas the product is equal to PSC. The equation (2.35) depicts that JSC uses both the sum and product of the two RRCs. Obviously, the measures CBS, JSC and PSC can completely be defined using RRC. However, MSC does not have such relation with RRC as can be seen from the equations (2.18) and (2.28).

(ii) From the expression (2.38), it is obvious that the value of JSC will be at least equal to PSC.

Further, both of these measures will be equal for those pairs of parts for which all the machines required by one part are also needed by the other part of the pair. In such situations, at least one of the two RRCs computed for two parts of a pair will be equal to one.

(iii) It should be noted that the relationship of RRC with CBS

shown in the equation (2.34) will also be valid when CBS as originally proposed by Studel and Ballakur (1987) is defined on the basis of processing times of the operations. However, RRC should also be defined on the basis of processing times.

(iv) The bound on the values of different commonality measures are as given below.

$$0 \le CBS_{i,j} \le 2 \qquad \forall\ i;\ \forall\ j$$

$$0 \le JSC_{i,j} \le 1 \qquad \forall\ i;\ \forall\ j$$

$$0 \le PSC_{i,j} \le 1 \qquad \forall\ i;\ \forall\ j$$

$$0 \le RRC_{i,j} \le 1 \qquad \forall\ i;\ \forall\ j$$

(v) Higher values of all these measures will indicate higher degree of closeness between a pair of parts.

(vi) The relationship between these measures will also hold for the case when parts have alternative process plans provided the definitions of CBS, JSC and PSC are suitably modified and extended.

(vii) The equations (2.29) and (2.34), and the equations (2.31) and (2.36) describing CBS and PSC on the basis of the number of machines can also be used to define them on the bases of the number of operations and the processing times. For these additional bases the matched requirement may not be symmetric. It may be recalled from the Section 2.2.1.5(f) that the matched requirement computed on the basis of the number of machines or tools is symmetric. Thus, the equation (2.35) which assumes symmetric property of the matched requirement, cannot be used for computing JSC. Further, the equation (2.30) cannot also be used for

this purpose because it is valid only when the requirements are expressed in terms of the number of machines. However, a measure on the basis of the number of operations and the processing times, that is quite similar and close to JSC, can be defined as:

$$JSC'_{i,j} = \frac{RM_{i,j} + RM_{j,i}}{R_i + R_j} \qquad \forall\ i;\ \forall\ j$$

(viii) It should be noted that the relationships of JSC with CBS and PSC given in the equations (2.37) and (2.38) will no longer be relevant if the measures CBS and PSC are computed on the basis other than the number of machines.

(ix) From the equations (2.29) and (2.31), it can be observed that the consideration of production volume will not have any effect on the values of the measures CBS and PSC defined on the bases other than even machines. The measure JSC (equation ((2.30)) whose definition is valid only for the number of machines, is not affected by the production volume since the consideration of the production volume does not alter the machine requirements of parts.

A measure on the basis of the number of machines or the processing times which incorporates the production volume and in close to JSC, can be defined as:

$$JSC''_{i,j} = \frac{w_i\ RM_{i,j} + w_j\ RM_{j,i}}{w_i R_i + w_j R_j} \qquad \forall\ i;\ \forall\ j$$

where w_i and w_j are the production volumes of the parts i and j, respectively.

2.4 COMPUTATIONAL REQUIREMENT

It can be seen from the equations (2.29), (2.30) to (2.31) and (2.34), (2.35) to (2.36) that while determining CBS, JSC and PSC, the measure RRC gets computed directly or indirectly at some intermediate step (directly in case of CBS and PSC, and indirectly for JSC). It can also be seen that, in essence, the values of the pair of RRCs are aggregated according to equations (2.34), (2.35) and (2.36) to give the values of the respective commonality measures. Since the process of aggregation will involve additional arithmetic operations, CBS, JSC and PSC will require more computations than RRC. Also as can be seen from the equations (2.18) and (2.28), MSC will require more computations than RRC.

In case of RRC, which need not be symmetric for pairs of parts, e.g. for parts i and j the pairs (i, j) and (j, i), a full matrix is required to contain the complete information about the closeness between the various parts, whereas for the other measures, triangular matrices will suffice. Moreover, the information available in the full matrix of RRC, showing the degree of closeness between a pair of parts with respect to each other, is more than the information contained in the triangular matrices of the other measures.

It should be noted that the triangular matrices of CBS, JSC, and PSC, as shown in the Section 2.3, can be determined from the matrix of RRC, and not vice versa.

From the discussions made in the above paragraphs, it can be concluded that RRC requires the least computation and contains more information.

2.5 DISCRIMINATING POWER

In order to analyze the relative usefulness of the commonality measures described in the Section 2.2 and 2.3, the discriminating characteristics of these measures need investigation.

Of course, the basic philosophy behind the use of the various commonality measures is the same, but the power of discriminating similar parts from dissimilar ones may be different. In order to make a comparison of the discriminating power of the various measures, the results and observations of the Section 2.3 will be used.

As shown in the previous sections, RRC as opposed to other measures gives the information about the closeness between a pair of parts using two distinct numbers each showing the degree of closeness of one part with respect to the other. In general, a better decision requires more as well as comprehensive information regarding the input parameters and rules. Thus, it can be expected that the RRC as compared to the other commonality measures will have better discriminating characteristic of identifying similar parts from dissimilar ones. However, from the definitions, expressions and the interrelationship, it is difficult to say, in general, about the relative discriminating power of the other measures.

On the basis of certain illustrations each depicting a different problem scenario, Chandrashekharan and Rajagopalan (1989) observe JSC to be better than MSC, and Ballakur and Studel (1987) observe CBS to be better than JSC for their discriminating characteristics.

Table 2.1: Commonality Measure Values for Different Problem Scenarios for the Comparison of the Discriminating Characteristics.

Scenario number	Number of machines for part i	for part j	common to parts i & j	uncommon to parts i & j	CBS_{ij}	JSC_{ij}	MSC_{ij} when the total number of machines are 2,000	4,000	10,000	PSC_{ij}	RRC_{ij}	RRC_{ji}
1	50	1000	50	950	1.0500	0.0500	0.5250	0.7625	0.9050	0.0500	1.0000	0.0500
2	100	1000	96	908	1.0560	0.0956	0.5460	0.7730	0.9092	0.0922	0.9600	0.0960
3	500	1000	350	800	1.0500	0.3043	0.6000	0.8000	0.9200	0.2450	0.7000	0.3500
4	1000	1000	525	950	1.0500	0.3559	0.5250	0.7625	0.9050	0.2756	0.5250	0.5250
5	100	50	50	50	1.5000	0.5000	0.9750	0.9875	0.9950	0.5000	0.5000	1.0000
6	150	150	100	100	1.3333	0.5000	0.9500	0.9750	0.9900	0.4444	0.6667	0.6667
7	200	250	150	150	1.3500	0.5000	0.9250	0.9625	0.9850	0.4500	0.7500	0.6000
8	350	250	200	200	1.3714	0.5000	0.9000	0.9500	0.9800	0.4571	0.5714	0.8000
9	1200	700	650	600	1.4702	0.5200	0.7000	0.8500	0.9400	0.5030	0.5417	0.9286
10	800	900	550	600	1.2986	0.4783	0.7000	0.8500	0.9400	0.4201	0.6875	0.6111
11	700	800	450	600	1.2054	0.4286	0.7000	0.8500	0.9400	0.3616	0.6429	0.5625
12	650	650	350	600	1.0769	0.3684	0.7000	0.8500	0.9400	0.2899	0.5385	0.5385
13	30	40	12	46	0.7000	0.2069	0.9770	0.9885	0.9954	0.1200	0.4000	0.3000
14	10	30	6	28	0.8000	0.1765	0.9860	0.9930	0.9972	0.1200	0.6000	0.2000
15	150	800	120	710	0.9500	0.1446	0.6450	0.8225	0.9290	0.1200	0.8000	0.1500
16	120	1000	120	880	1.1200	0.1200	0.5600	0.7800	0.9120	0.1200	1.0000	0.1200
17	900	900	600	600	1.3333	0.5000	0.7000	0.8500	0.9400	0.4444	0.6667	0.6667
18	660	660	440	440	1.3333	0.5000	0.7800	0.8900	0.9560	0.4444	0.6667	0.6667
19	450	450	300	300	1.3333	0.5000	0.8500	0.9250	0.9700	0.4444	0.6667	0.6667
20	120	120	80	80	1.3333	0.5000	0.9600	0.9800	0.9920	0.4444	0.6667	0.6667
21	10	30	0	40	0.0000	0.0000	0.9800	0.9900	0.9960	0.0000	0.0000	0.0000
22	10	30	10	20	1.3333	0.3333	0.9900	0.9950	0.9980	0.3333	1.0000	0.3333

In order to judge the relative discriminating power of CBS, JSC, MSC, PSC and RRC, a set of 22 different scenarios for the two parts of a pair is considered. The parameters of the problem scenarios and the various relevant computed values are presented in Table 2.1. The scenarios are grouped into six blocks (see Table 2.2). In blocks I to V, at least one of the commonality measure has a constant value, whereas block VI contains the scenarios with extreme conditions showing no matching to complete matching of a part with respect to the other. In each block, scenarios included represent a wide range of number of common machines and total machine requirements for the pair of parts. In addition, for the scenarios in each block, the ratio of the number of the common machines to the total machine requirement is taken to be different.

Table 2.2: Blocks of Scenarios Given in Table 2.1.

Block	Scenarios	Measures which are constant
I	1, 2, 3, 4	CBS
II	5, 6, 7, 8	JSC
III	9, 10, 11, 12	MSC
IV	13, 14, 15, 16	PSC
V	17, 18, 19, 20	CBS, JSC, PSC, RRC
VI	21, 22	-

Following are the discussions made on the basis of the values of given parameters and computed commonality measures shown in the Table 2.1. These results are also used for validating the observations regarding the relations between numerical values of

the commonality measures shown and discussed in the previous sections.

(i) It can be observed from the scenarios of Block V that when $RRC_{i,j}$ and $RRC_{j,i}$ are constant, the measures $CBS_{i,j}$, $JSC_{i,j}$ and $PSC_{i,j}$ are also constant. This is so because $CBS_{i,j}$, $JSC_{i,j}$ and $PSC_{i,j}$ can be uniquely determined from $RRC_{i,j}$ and $RRC_{j,i}$ (see the equations (2.34), (2.35) and (2.36)). Therefore, it can be said that for the situations where the ratios of the number of the common machines to the total machine requirements of the two parts of a pair are constant, RRC along with CBS, JSC and PSC are also constant.

(ii) For each of the scenarios, the values of JSC are greater than or equal to PSC. This is in conformance to the expression (2.38). Further, as can be seen from the scenarios 1, 5, 20 and 22, whenever $JSC_{i,j}$ and $PSC_{i,j}$ are equal, at least one of the values of $RRC_{i,j}$ and $RRC_{j,i}$ is equal to one.

(iii) As seen from the Tables 2.1 and 2.2, while CBS, JSC and PSC remain constant corresponding to the scenarios of the Blocks I, II and IV, the RRC shows variations in its values. As mentioned earlier, the values of these measures are also constant whenever RRC is constant. Thus, it can be concluded that the CBS, JSC and PSC as compared to RRC are less sensitive to recognize the variations in degree of closeness between pairs of parts.

For example, in the scenarios of the Block IV, when PSC has a constant value of 0.1200, indicating the same degree of closeness between the parts i and j in all the scenarios, the values of $RRC_{i,j}$ range from 0.4000 to 1.000, and those of

$RRC_{j,i}$ from 0.1200 to 0.3000. In scenario 13, out of the total machine requirements (henceforth to be termed as requirement only) of 30 for part i, and that out of 40 for part j, 12 machines are common. However, in the scenario 16, there are 120 common machines out of the total requirement of 120 for part i and out of 1000 for part j. The measure PSC remains insensitive to these distinct numbers and proportions of the common machines. But the measure RRC reflects these variations by having distinct values. Similar observations about the insensitivity of the CBS and JSC can be made from the scenarios of the Blocks I and II, respectively.

(iv) In the scenarios 1 and 2, most of the requirements of part i match with that of part j. In the scenario 1, all the 50 machines required by part i are also needed by part j, whereas in the scenario 2, 96 machines out of 100 required by part i are also needed by part j. In these two scenarios, part i stands as a strong candidate to belong to the group of part j. However, for these two scenarios both $JSC_{i,j}$ and $PSC_{i,j}$ take small values ($JSC_{i,j}$ = 0.0500 and 0.0956, and $PSC_{i,j}$ = 0.0500 and 0.0922) which may discourage the grouping of parts i and j. The values of these measures are small because of the poor matching requirements of part j with that of part i, and also due to aggregation of the matching requirements of the part i with j and that of j with i. Such a probable error in grouping decision can be avoided by not aggregating, but using the matching requirements of parts computed separately with respect to each other. The introduction of the concept of RRC is an attempt in this direction.

The values of RRCs shown in the Table 2.1, which distinctly recognize the individual matching requirements, can be used for such purposes.

(v) It can be seen from the Table 2.1 that whenever JSC and PSC take somewhat high values, the value of CBS (at the scale of 0.00 to 2.00) is also reasonably high. For example, in the scenario 8, the value of $CBS_{i,j}$ is 1.3714, while $JSC_{i,j}$ and $PSC_{i,j}$ are 0.5000 and 0.4571, respectively.

However, as can be seen for the pairs of parts (i, j) of the scenarios 1, 2, 5, 16 and 22, where compatibility of the part i (j) with respect to j (i) is high and low for the part j (i) with respect to i (j), JSC and PSC take small values though the value of CBS remains reasonably good. In general, if the value of any commonality measure for a pair of parts is high, then the chances of grouping these two parts together will be better; and vice versa. Therefore, for these scenarios representing extreme situations where a part is almost compatible with the other part, the smaller values of JSC and PSC will discourage the grouping of these parts together, whereas using CBS, one can have high hopes for their grouping. The reason for such discrepancy is as follows.

For a pair of parts, in the above mentioned extreme situations, one of the RRC will be high and the other will be small. Since RRC can at most be equal to one, the multiplication of these two fractional values of RRCs will result a smaller number. Therefore, in this case, JSC and PSC which use the product of two RRCs (see equations (2.35)

and (2.36)), will also be small. However, CBS which is the sum of the two RRCs will be reasonably good.

From the above discussions, it can be said that CBS as compared to JSC and PSC will have better discriminating characteristics. This is further illustrated through numerical examples in Section 2.7.

(vi) As illustrated by the scenarios of Block I, $CBS_{i,j}$ fails to differentiate the degree of matching of the requirements of part i with part j which varies from 50 common machines out of 50 machines total required to 525 common machines out of 1000 total required by part i. This, again, is due to the aggregation of matching requirements of parts i and j. The values of RRCs as given in Table 2.1 do remain sensitive to such variation.

(vii) From the various scenarios, it can be seen that MSC changes with variations in the number of uncommon machines and also with the total number of machines considered for the grouping. Further, as can be seen from the scenarios of the Block III that when the number of uncommon machines remains constant, the value of MSC is also constant. Thus, MSC is observed to be insensitive for recognizing the variations in the number of machines commonly required by individual parts of a pair and its proportion to their total requirement.

For example, in the scenario 1, the value of $MSC_{i,j}$ is 0.5250 when the number of uncommon machines are 950 and the total number of machines considered for grouping is 2000. The value increases to 0.9050 when the total number of machines considered for grouping becomes 10,000. The value

of $MSC_{i,j}$ does not change for the scenario 4 for which the number of uncommon machines remains the same. However, for the scenario 3, when the number of uncommon machines is 800, the value becomes 0.6000.

Further, in the scenario 21, though the requirements of machines of parts i and j are totally different, $MSC_{i,j}$ is quite high and so will be the chance of grouping these two parts together. Interestingly enough, when the part i becomes fully compatible with part j, improvement in $MSC_{i,j}$ value is insignificant as can be seen from the scenario 22. However, the other measures, initially at zero in the scenario 21, increase remarkably with increase in the number of common machines as shown in the scenario 22. In the scenario 1, requirements of part i are completely compatible with that of part j; whereas in the scenario 21, the requirements of the two parts are completely different. However, $MSC_{i,j}$ for the scenario 1 is significantly lower than for the scenario 21 when the total number of machines considered for grouping is 2000. Obviously, the MSC value may be quite high for dissimilar parts and low for the parts with high matched requirements.

The reason of occurrence of the above problems can be attributed to the method of computing the MSC. The MSC ignores the individual requirements of parts and depends upon the number of uncommon machines between the pairs of parts and the total number of machines considered for grouping.

Since the aforementioned problems do not arise with other commonality measures as can be seen from the scenarios

given in the Table 2.1, it can be said that the other measures will possess better discriminating characteristics than the MSC.

From the discussions and observations made in the above paragraphs regarding the discriminating characteristics of the commonality measures, it can be concluded that the RRC, as compared to other measures, has the highest sensitivity to capture the variations in the problem parameters and thus has higher capability of identifying similar parts from dissimilar ones.

2.6 COMMONALITY MEASURES AND THEIR NUMERICAL VALUES

As mentioned earlier, various commonality measures (CBS, JSC, PSC and RRC) depend upon the matching requirements of parts of a pair, and for a given pair resume different values according to their definition. A measure may have the same numerical value for a number of pairs of parts having different combinations of the problem parameters. This has been seen already in the Table 2.1 where a measure has the same value in a block containing a variety of scenarios. These open the following issues related with the numerical values of the commonality measures for further investigation.

(a) How sensitive the values of commonality measures are for capturing correctly the variations in the parameters of various problem situations?

(b) What is the nature of the variation in the values of the commonality measures for the whole range of the combinations of the matching of two parts of a pair?

(c) What value of a commonality measure can be considered to be good (or bad) for showing reasonable closeness (or

dissimilarity) between the two parts of a pair, and for the purpose of grouping in general?

The issue, as mentioned in (a) above, related with the sensitivity of the commonality measures has been discussed in detail in the Section 2.5 using their mathematical relationships and the various scenarios illustrated in the Table 2.1. However, it is not possible to investigate the issues (b) and (c) with these aids. For this purpose, a set of 1,000,000 pairs of parts covering a wide spectrum of different levels of matching in the machine requirements of the two parts of these pairs, are generated and the frequency distributions of the commonality measures are collected.

The level of the matching, relating the number of machines common to both the parts of a pair with their individual total machine requirement is the same as relative requirement compatibility of the two parts computed with respect to each other. Therefore, the frequency distribution showing different levels of matching in the machine requirements will be the same as that for the values of RRC. In other words, the distribution of RRC values itself will represent the distribution for the matching of the requirements. Henceforth, the reference to the distribution of the level of matching will be made by that of RRC.

The values of the RRCs for the two parts of each of these pairs are generated using random number generator of IMSL library, and then mathematical expressions (2.34), (2.35) and (2.36) of the Section 2.3 are used to compute the values of CBS, JSC and PSC from the two RRCs. Some ten distributions allowing a variety of combinations of problem parameters as reflected by the matching of

the machine requirements are considered to study the behaviour of the values of various commonality measures. The distributions are as shown in Table 2.3. The distributions I and II represent

Table 2.3: Distributions for Matching of Machine Requirements.

Distribution	Description
I	Normal (mean = 0.25, std. dev. = 0.0888)
II	Beta (α = 1.1, β = 3.3)
III	Beta (α = 0.5, β = 1.0)
IV	Beta (α = 0.9, β = 0.9)
V	Uniform (a = 0.0, b = 1.0)
VI	Triangular (symmetric with mean = 0.5)
VII	Normal (mean = 0.5, std. dev. = 0.1666)
VIII	Beta (α = 1.0, β = 0.5)
IX	Beta (α = 3.3, β = 1.1)
X	Beta (α = 5.5, β = 1.1)

situations where parts are mostly dissimilar from each other and thus are not so suitable for grouping. The distributions VIII, IX and X, on the other hand, represent situations where most of the parts are quite close to each other and thus may be required to be put into few groups only. The other distributions, of course, represent the general situations and are normally considered for grouping.

The resulting frequency and cumulative distributions for these measures are shown in Figures 2.1 to 2.10, and the other relevant statistics in Table 2.4. Since the interval for CBS value is [0,2] instead of [0,1] as for the rest of the other measures, for a better understanding of the variability in the values of the commonality measures, their interval lengths (R) are normalized and expressed in terms of their respective standard

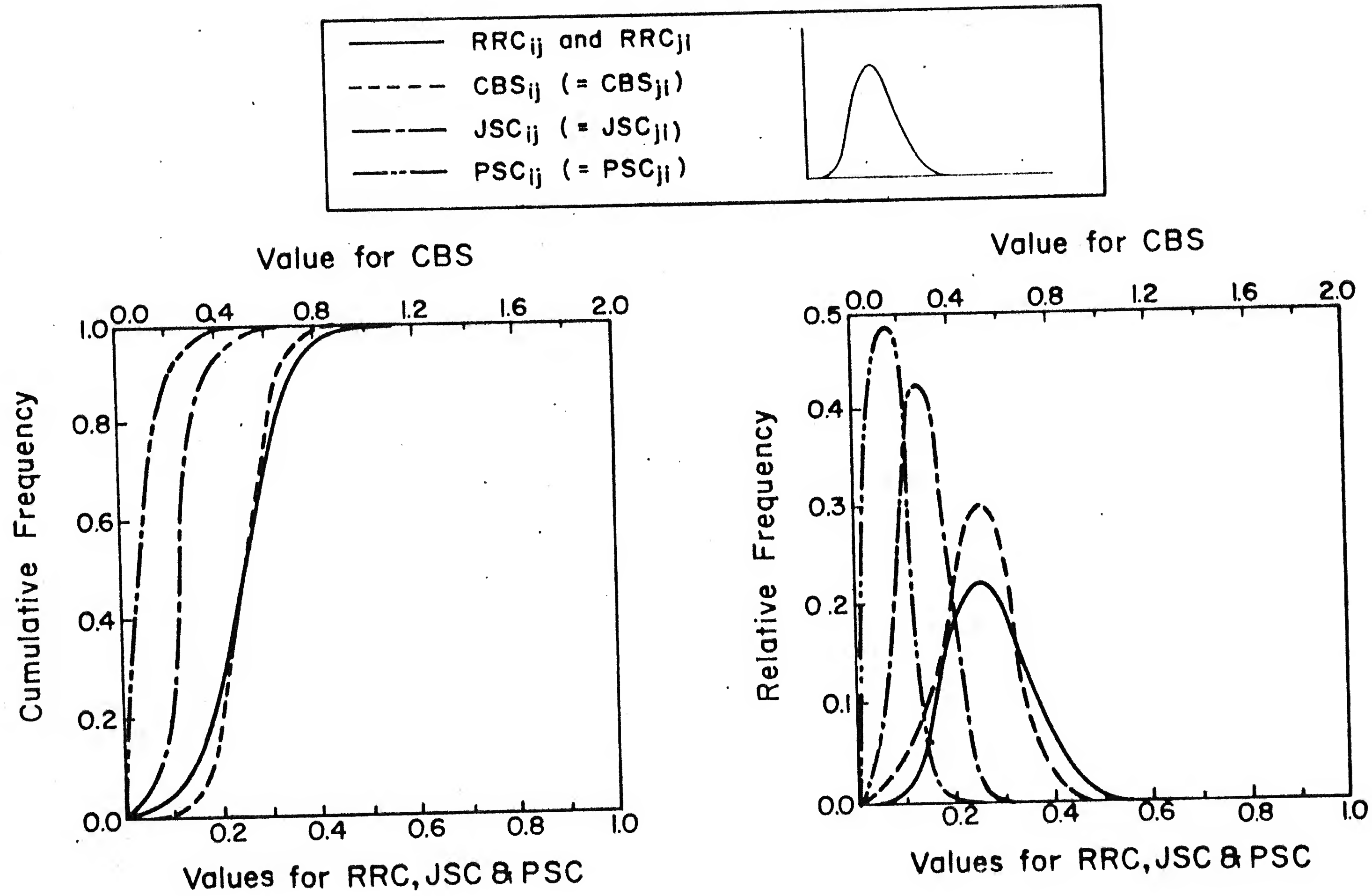


Figure.2.1: Frequency Density and Cumulative Distribution Curves for Matching Distribution : Normal ($\mu = 0.25$, $\sigma = 0.0888$).

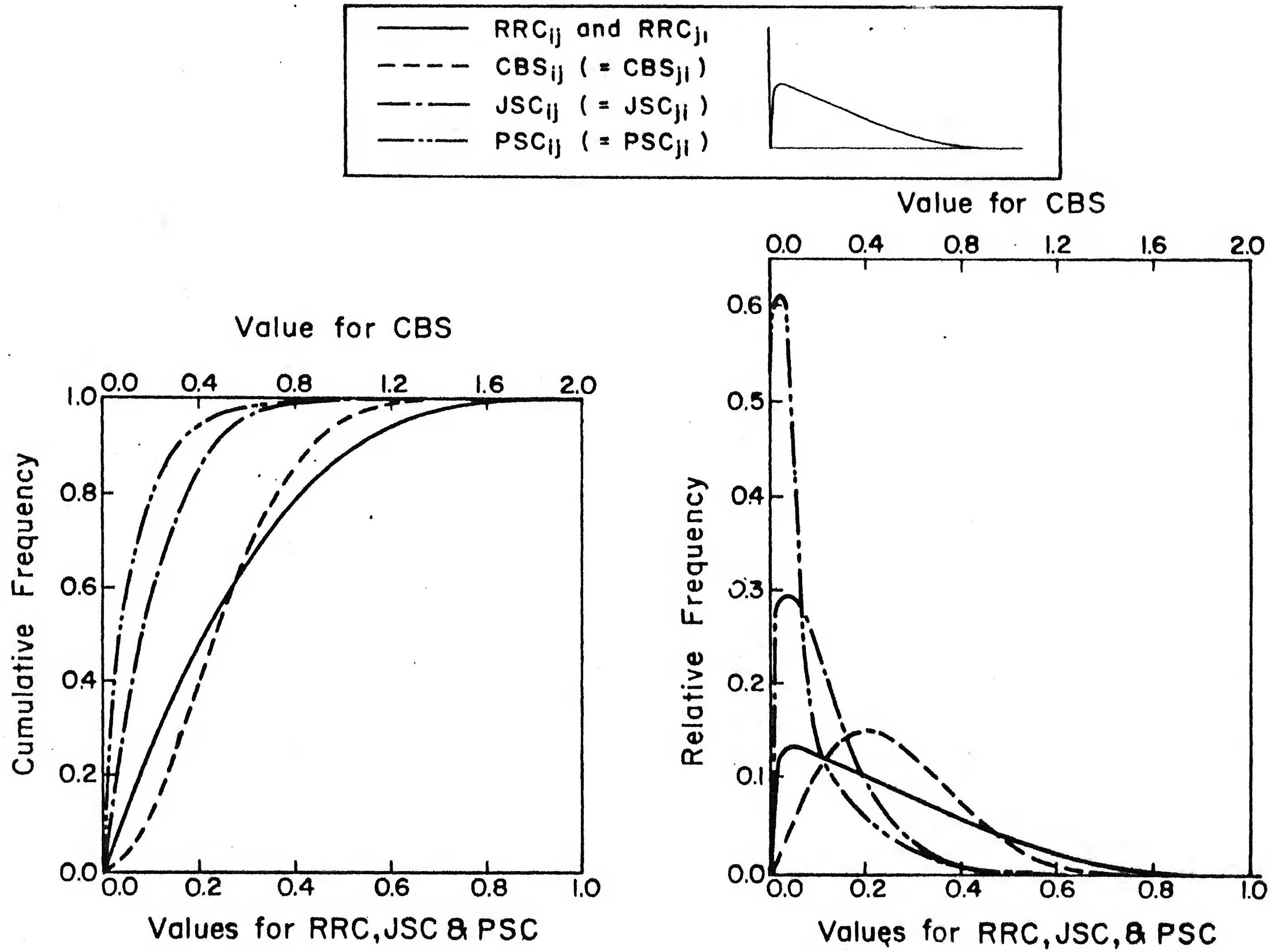


Figure.2.2: Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta ($\alpha = 1.1, \beta = 3.3$).

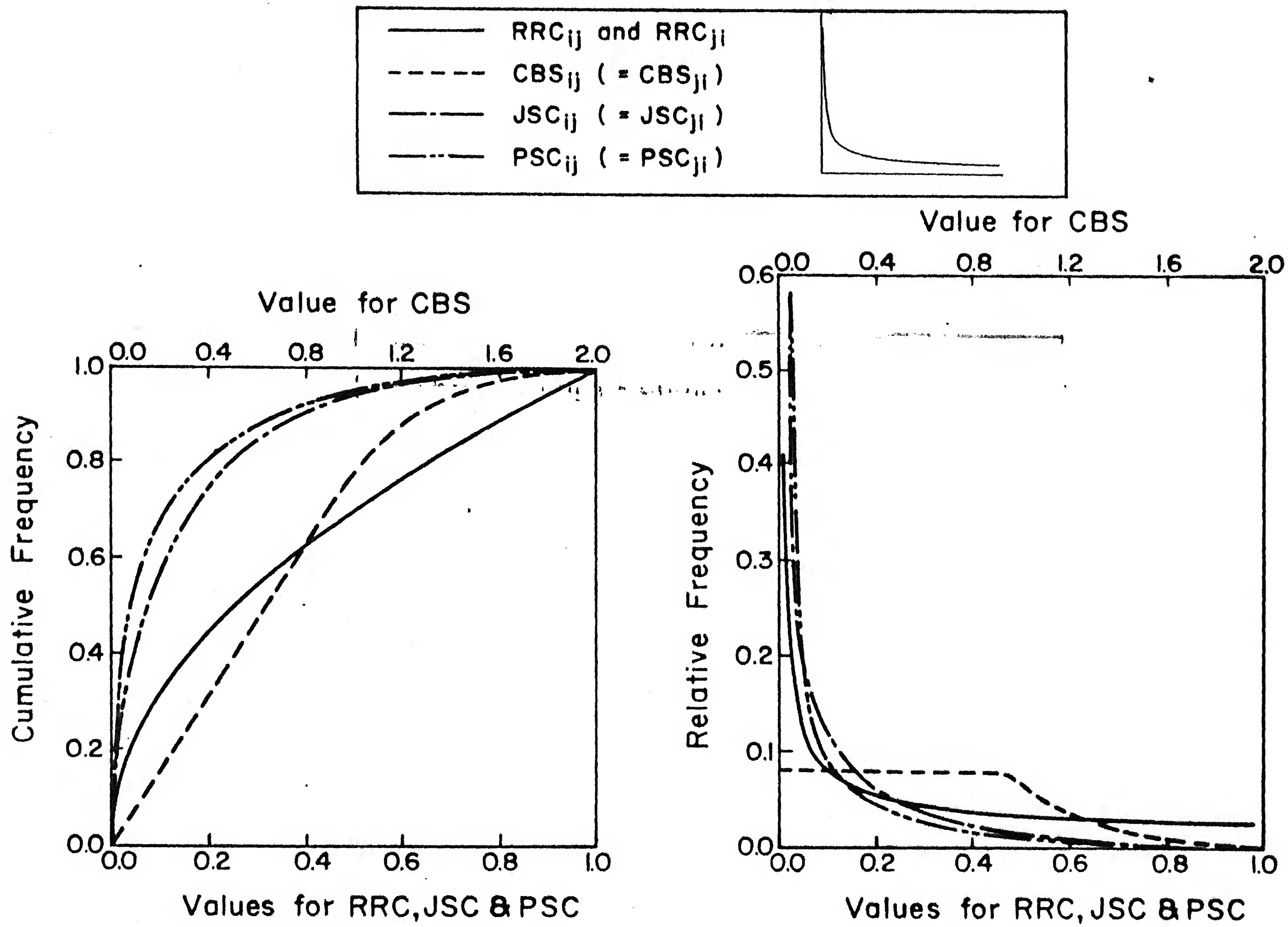


Figure 2.3: Frequency Density and Cumulative Distribution Curves for Matching Distribution : Beta (α = 0.5, β = 1.0).

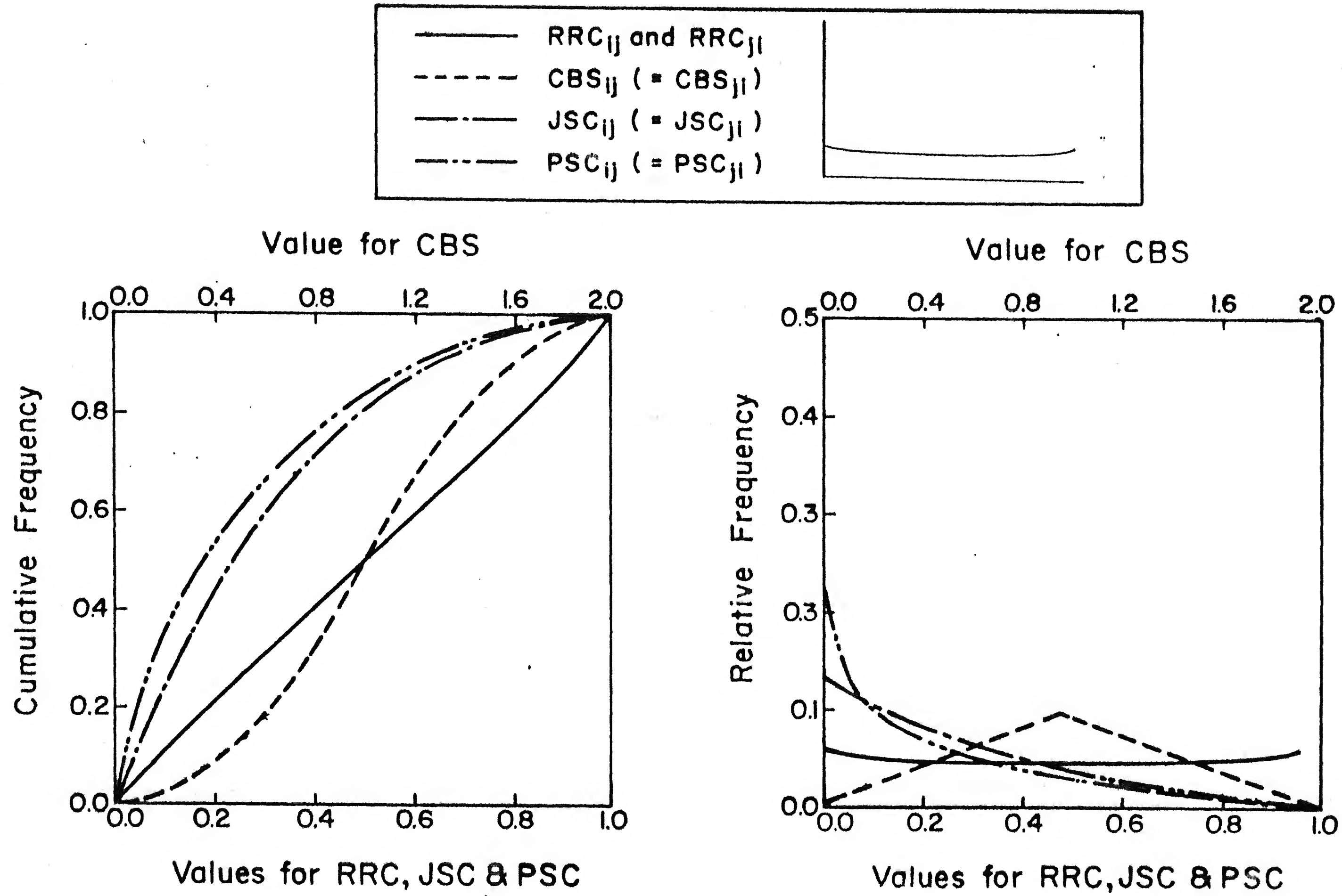


Figure.2.4: Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta (α = 0.9, β = 0.9).

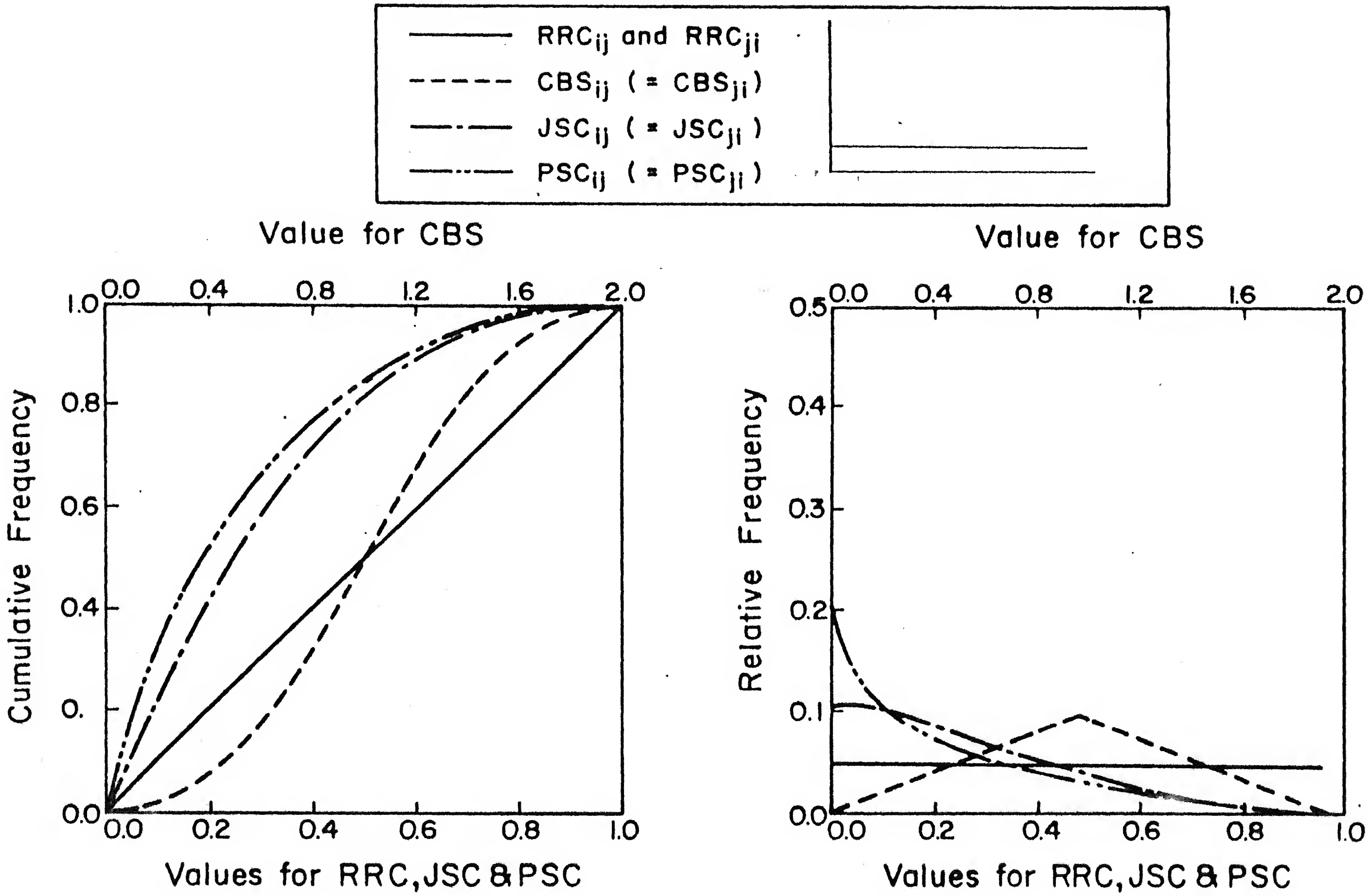


Figure.2.5: Frequency Density and Cumulative Distribution Curves for Matching Distribution : Uniform (a = 0.0, b = 1.0).

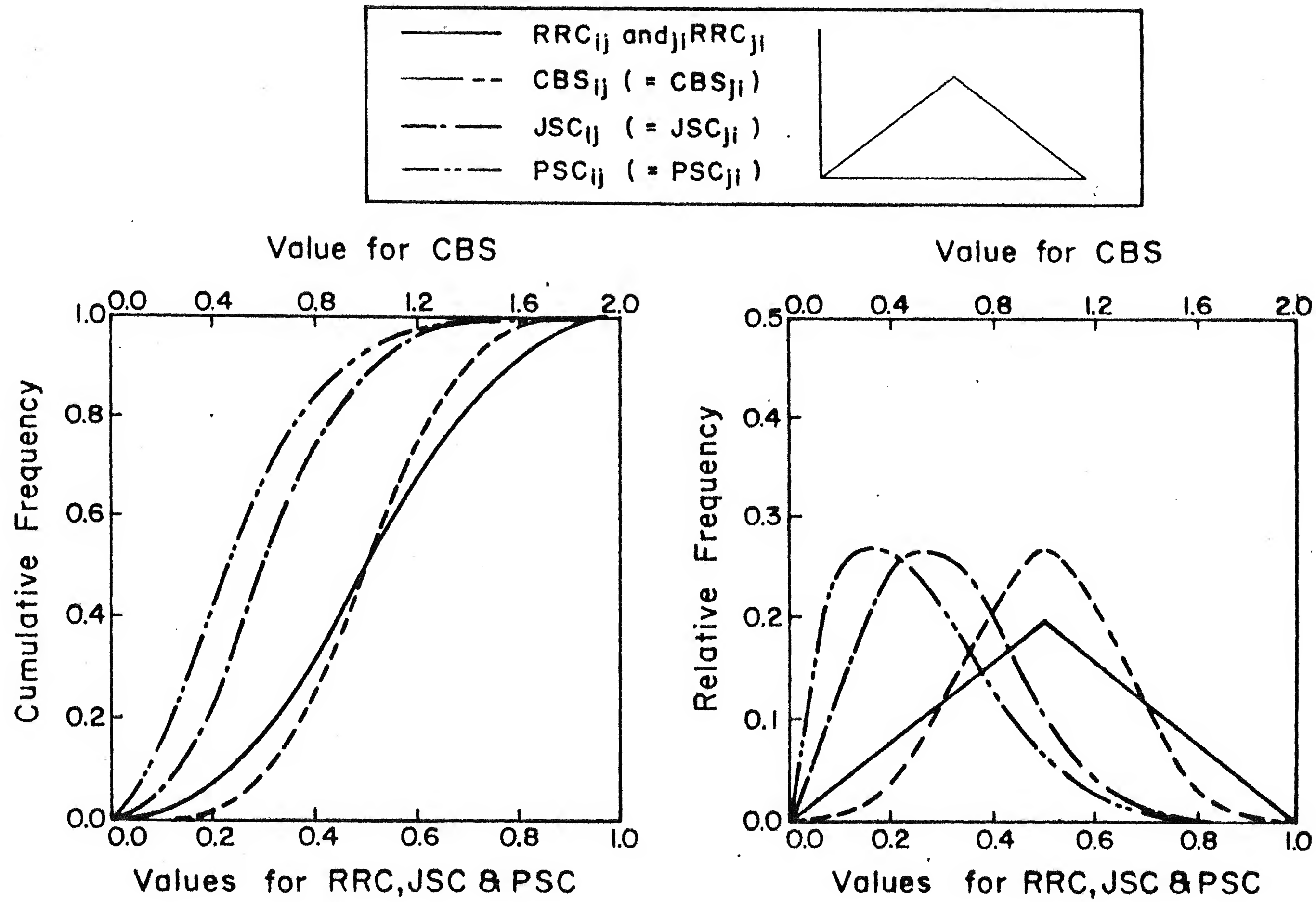


Figure. 2.6: Frequency Density and Cumulative Distribution Curves for Matching Distribution: Triangular (Symmetric with mean = 0.5).

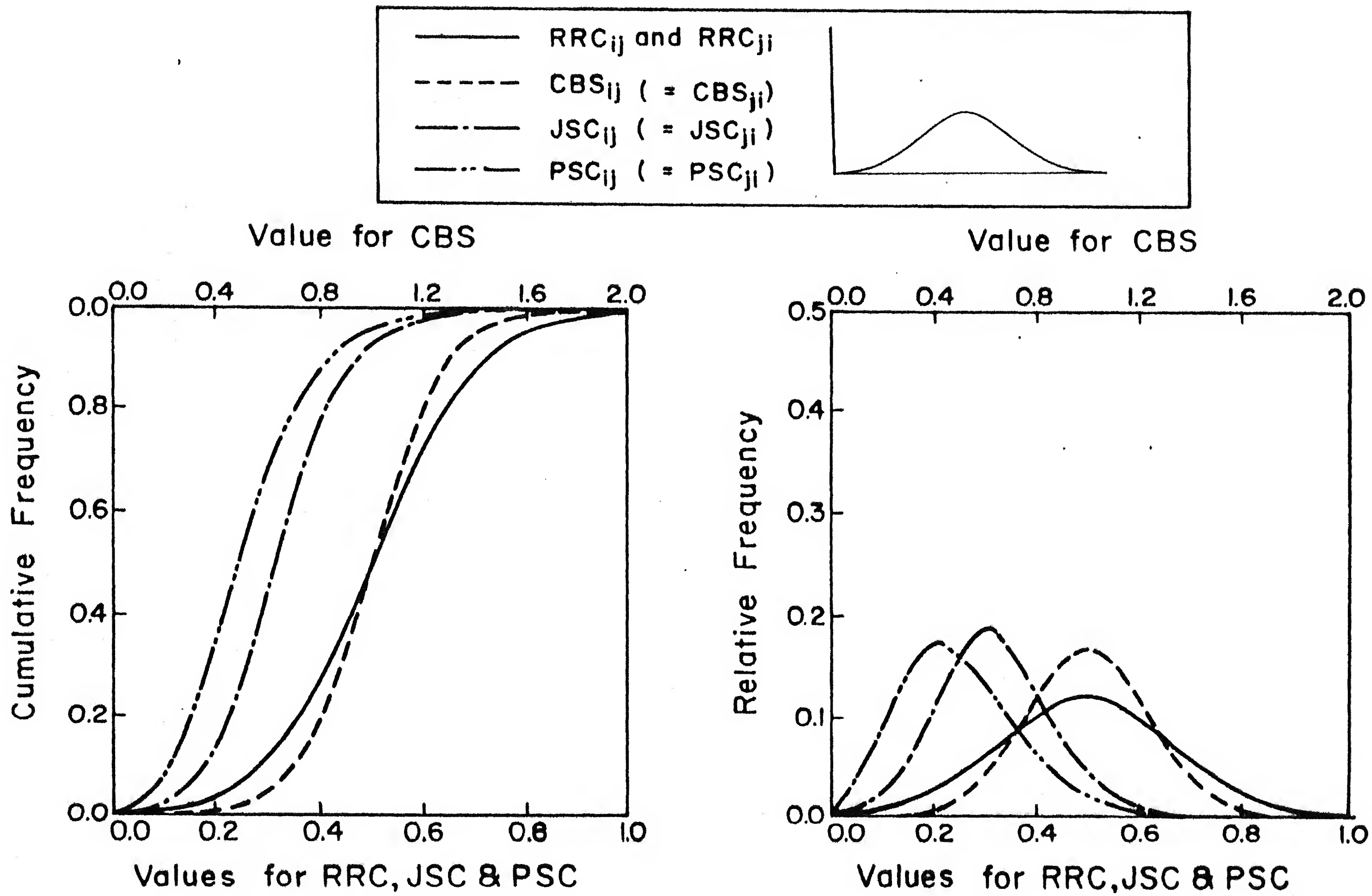


Figure.2.7: Frequency Density and Cumulative Distribution Curves for Matching Distribution: Normal ($\mu = 0.5$, $\sigma = 0.1666$).

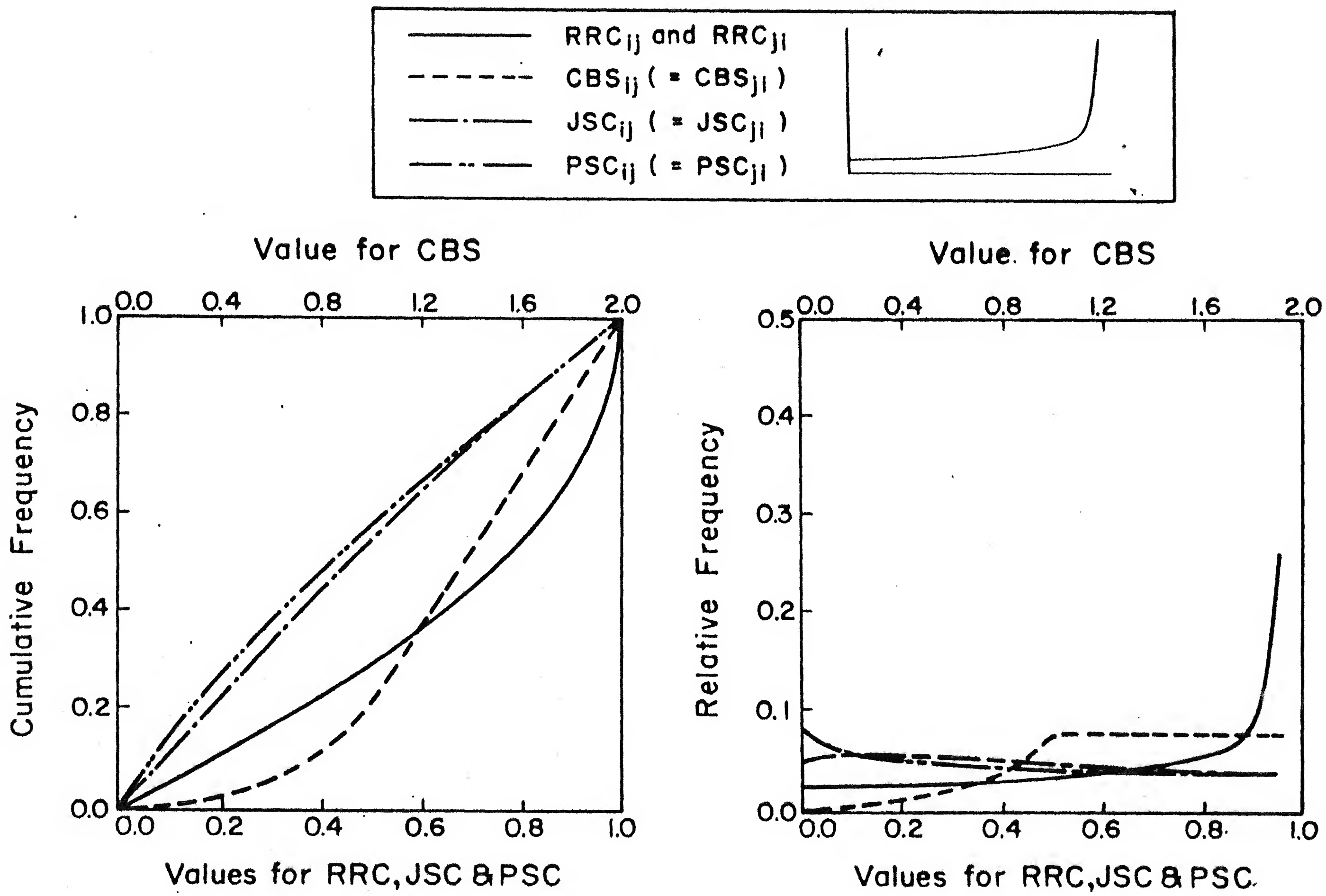


Figure. 2.8: Frequency Density and Cumulative Distribution Curves for Matching Distribution: Beta (α = 1.0, β = 0.5).

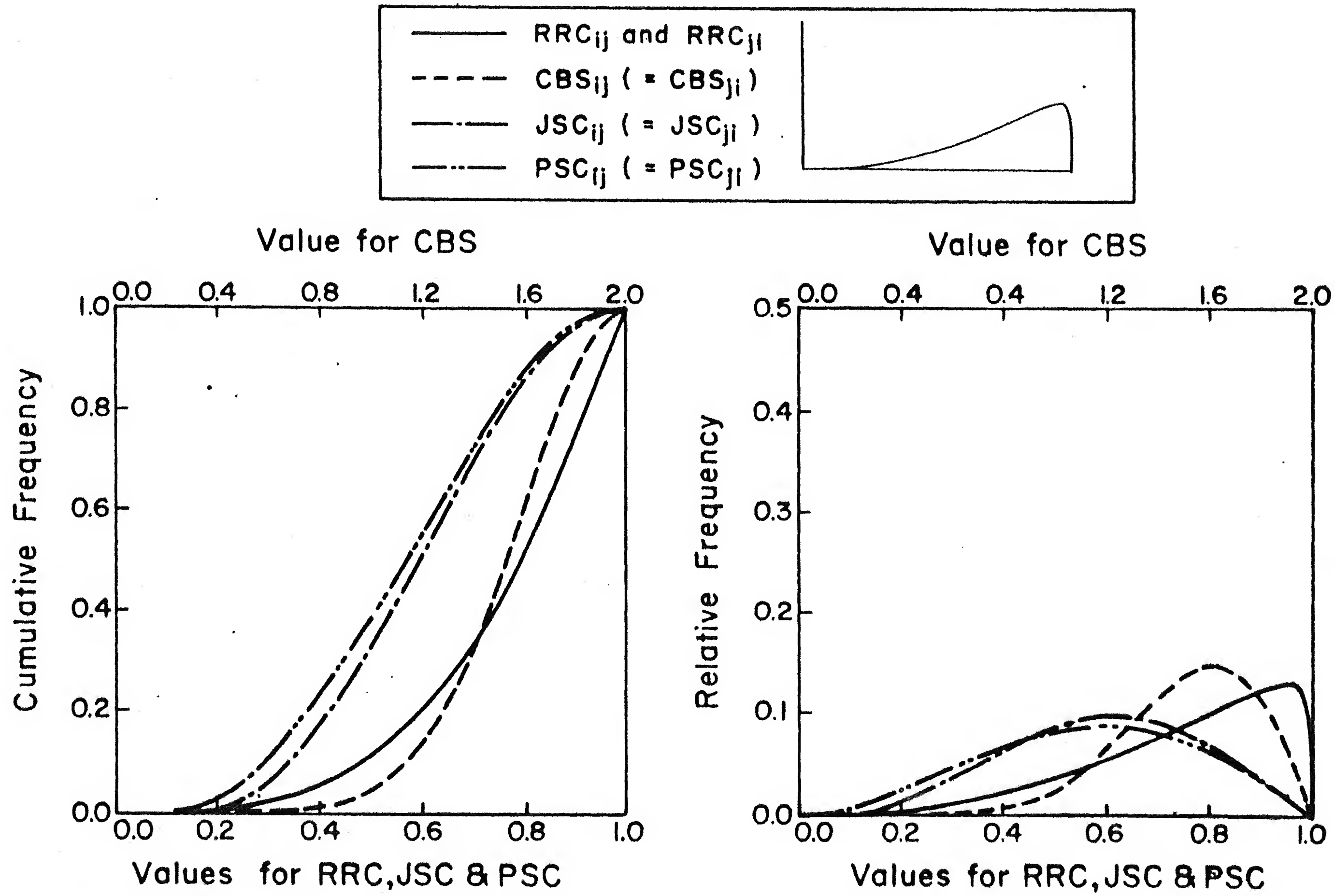


Figure.2.9: Frequency Density and Cumulative Distribution Curves for Matching Distribution : Beta ($\alpha = 3.3$, $\beta = 1.1$).

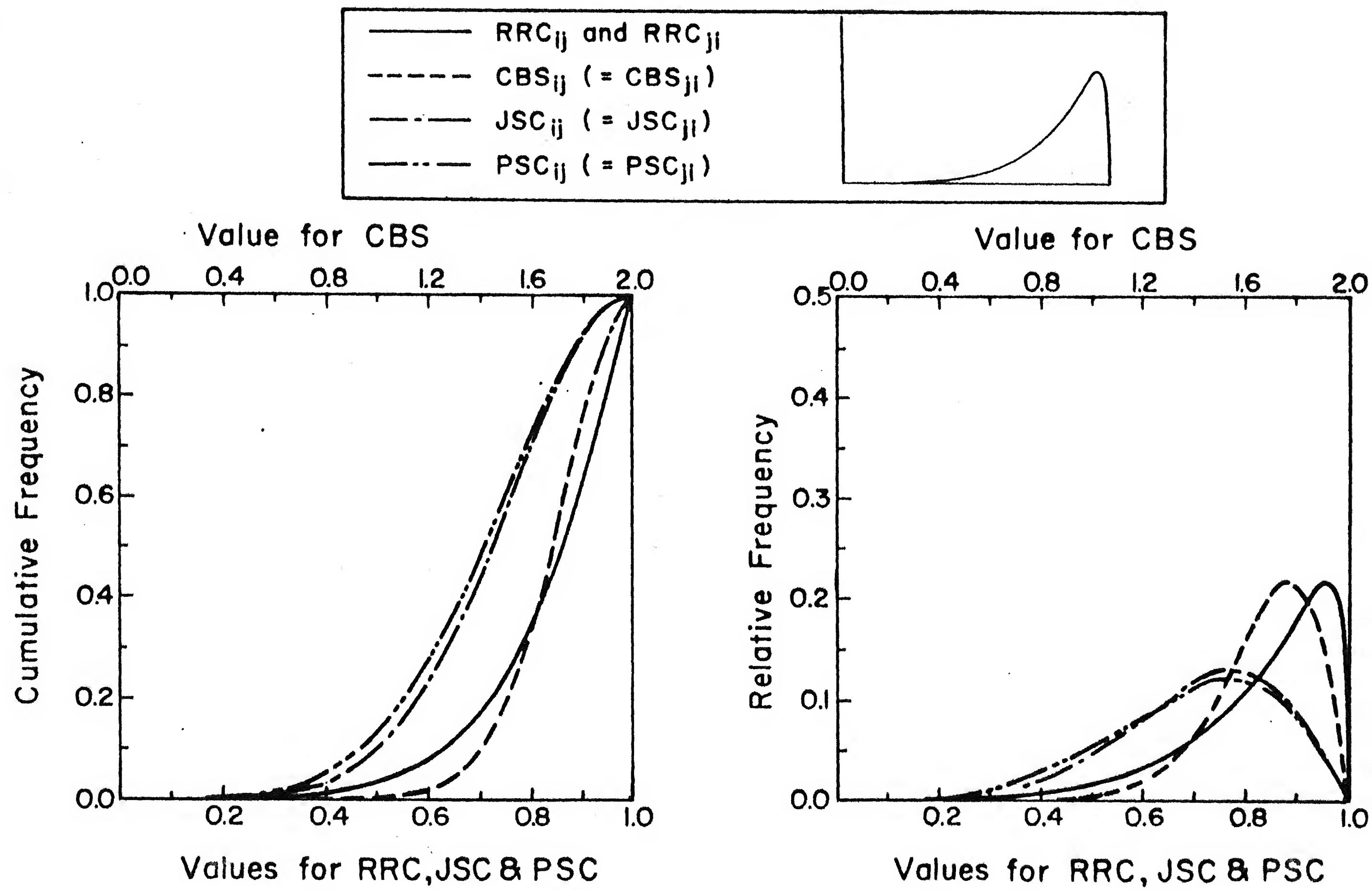
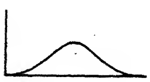


Figure.2.10: Frequency Density and Cummulative Distribution Curves for Matching Distribution: Beta (α = 5.5, β = 1.1).

Table 2.4 : Some Related Statistics on Various Commanality Measures for Different Matching Requirement Distribution.

Distribution		RRC			CBS			PSC			JSC		
Shape	Number												
	I	0.2500	0.0827	12.0881	0.5005	0.1169	17.1054	0.0626	0.0300	33.2898	0.1351	0.0432	23.1304
	II	0.2501	0.1863	5.3666	0.5003	0.2630	7.6040	0.0624	0.0741	13.5048	0.1108	0.0901	11.0983
	III	0.3330	0.2979	3.3564	0.6659	0.4219	4.7410	0.1111	0.1665	6.0073	0.1416	0.1737	5.7572
	IV	0.5000	0.2986	3.3491	0.9999	0.4223	4.7355	0.2500	0.2292	4.3637	0.2873	0.2244	4.4561
	V	0.4998	0.2888	3.4628	0.9997	0.4086	4.8952	0.2499	0.2206	4.5338	0.2898	0.2154	4.6422
	VI	0.5000	0.2041	4.8991	1.0000	0.2885	6.9328	0.2500	0.1501	6.6600	0.3100	0.1431	6.9857
	VII	0.4992	0.1643	6.0897	0.9981	0.2327	8.5944	0.2492	0.1191	8.3948	0.3170	0.1120	8.9276
	VIII	0.6665	0.2982	3.3530	1.3331	0.4236	4.7215	0.4450	0.2953	3.3866	0.4681	0.2839	3.5219
	IX	0.7505	0.1859	5.3778	1.5010	0.2633	7.5974	0.5633	0.2005	4.9882	0.5897	0.1843	5.4261
	X	0.8335	0.1350	7.4064	1.6670	0.1910	10.4731	0.6947	0.1602	6.2431	0.7102	0.1478	6.7671

deviations (σ). A higher value of the ratio R/σ will indicate a lower variability for the corresponding commonality measure.

In the following subsections, some efforts are made to answer the questions (a), (b) and (c) raised above.

2.6.1 Sensitivity of Values for Capturing Variations in Problem Scenarios

It can be seen from the frequency distribution curves shown in the Figures 2.1 to 2.10 that the distribution curves for CBS closely follow the pattern of RRC which also represent the variations in the matching requirements, whereas those of JSC seem to be somewhat away, and those of PSC comparatively more away. This fact can also be noticed from the cumulative distribution curves. The curves for CBS are closest to that of RRC, whereas of JSC farther, and that of PSC are farthest from the corresponding cumulative curves of RRC.

In general, the numerical value of a commonality measure should itself provide sufficient idea about the kind and amount of closeness, or in other words, the level of matching in machine requirements of the two parts of pair. As observed in the previous paragraph, the measures other than RRC do not have a strong property of this kind and also the power of identifying the true variation in the closeness of parts is not the same for these measures. Thus the measures, based on the decreasing tendency of capturing the true variation in the level of matching in the machine requirements, can be sequenced as RRC, CBS, JSC and PSC.

Since the discriminating power of a commonality measure is characterized by its ability to provide the true picture of

closeness, the above sequence represents the commonality measures in the order of their decreasing power of discrimination.

2.6.2 Nature of Variations in the Values of Commonality Measures

From the frequency distribution curves shown in the Figures 2.1, 2.4, 2.5, 2.6 and 2.7, it can be seen that whenever the distribution for RRC is symmetric, it is also symmetric for CBS. For all the cases, the frequencies at and around the mean value are more for CBS compared to that for RRC. From the Table 2.4, it can be seen that for each of the distributions the mean of the RRC (μ_{RRC}) is half of that for CBS (μ_{CBS}). However, the ratios of mean values of CBS and RRC to their respective interval lengths are equal.

The ratio of interval length to standard deviation (R/σ) for CBS being always greater than for RRC indicates that the variability in the values of RRC, in general, will be higher than that for CBS. For the distributions IV, V, VI and VII, the μ_{RRC} and μ_{CBS} are almost constant and are around 0.5 and 1.000, respectively. But the respective values of R/σ for RRC are 3.3491, 3.4628, 4.8991 and 6.0877, whereas those for CBS are 4.7355, 4.8952, 6.9328 and 8.5944, respectively. These values clearly show that the variability in the value of CBS decreases with decrease in the variability of the values of RRC. The same characteristic can be observed for the distributions I and II. For the distributions III, IV and VIII, the values of μ_{RRC} are not the same, but σ_{RRC} is the same and is around 0.3. Similarly, for these distributions, the values of μ_{CBS} are also not constant and σ_{CBS} is constant with its value around 0.42. Similar observations can be made from the distributions II and IX. Further, from

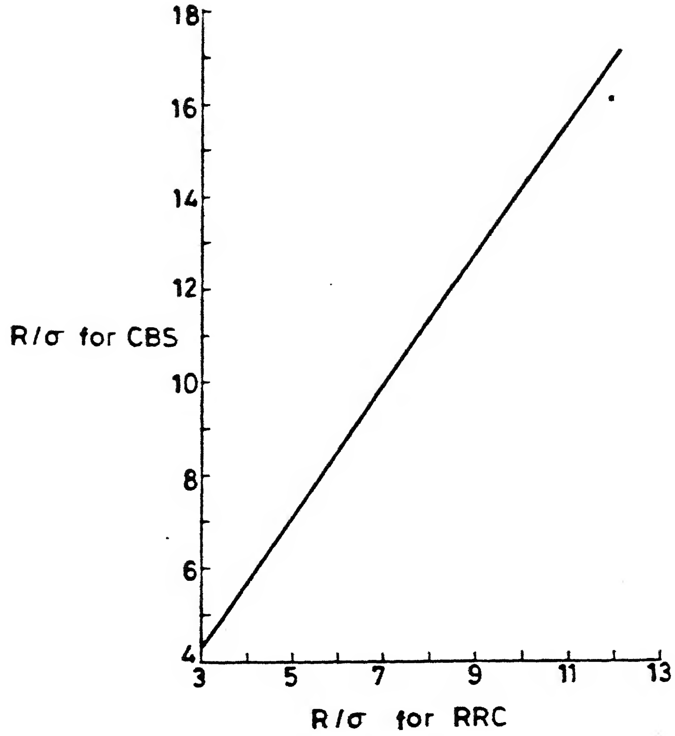


Figure 2·11: Relationship Between the Ratio of Interval Length and Standard Deviation (R/σ) for RRC and CBS.

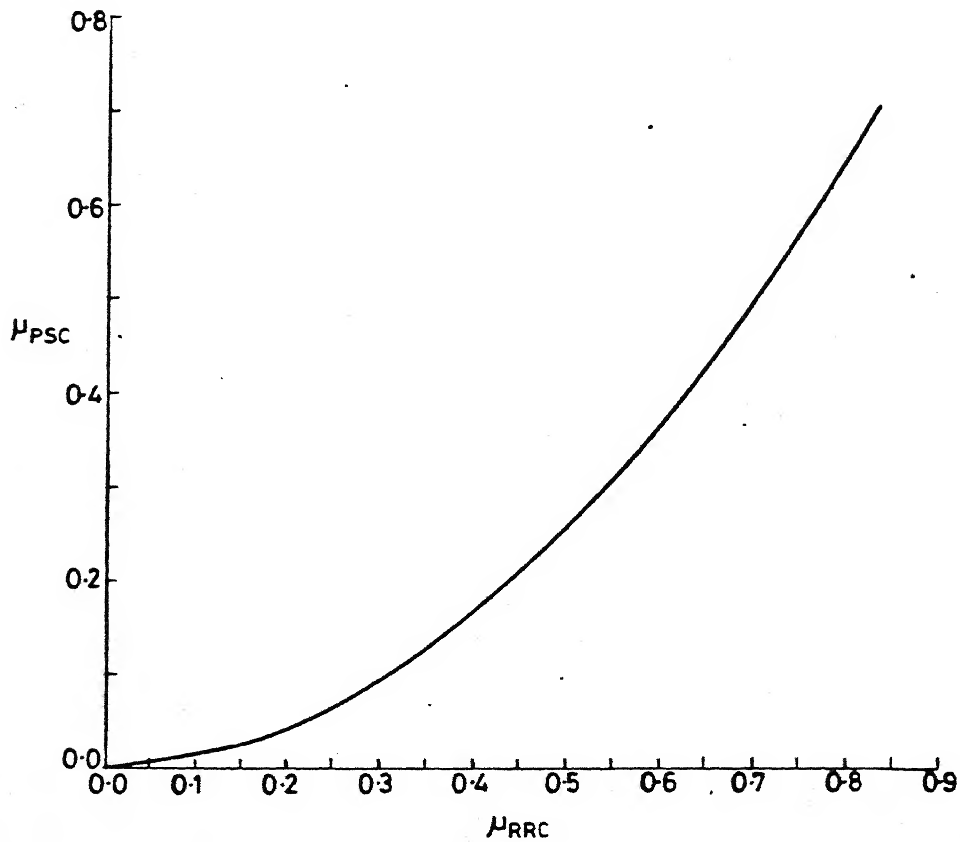


Figure 2·12: Relationship Between the Mean Values of RRC and PSC.

Figure 2.11, it can be seen that R/σ for CBS in monotonically increasing function of R/σ of RRC. In addition, σ_{CBS} is found to be almost 1.414 times of σ_{RRC}.

The relations observed above regarding mean and R/σ values of CBS and RRC are also found supported with analytical results as discussed below.

Let R1 and R2 are two independent and identically distributed random variables denoting two RRCs corresponding to the parts 1 and 2 forming a pair and computed with respect to each other, and corresponding value of CBS be C. Then, from the equation (2.34),

$$C = R1 + R2.$$

Since R1 and R2 are independent and identically distributed random variables, the mean (μ_C), and the standard deviation (σ_C) of variable C will be:

$$\mu_C = 2\mu_R \tag{2.39}$$

and

$$\sigma_C^2 = 2\sigma_R^2, \tag{2.40}$$

where μ_R and σ_R are the mean and standard deviation of RRC. The relationship between means and standard deviations of RRC and CBS obtained from the simulation are the same as shown by the equations (2.39) and (2.40). From this result the validity of the simulation can be confided.

The kind of relations which exist between means and standard deviations of CBS and RRC do not apply to the means and standard deviations of any other pair of the commonality measures.

Looking into the values of μ_{RRC} and μ_{PSC} for the distributions I and II, and also for the distributions IV, V, VI and VII,

it can be said that μ_{PSC} is directly related with μ_{RRC}. As long as μ_{RRC} is constant, μ_{PSC} also remains constant even though σ_{RRC} may vary. The relation between μ_{RRC} and μ_{PSC} is shown in Figure 2.12. It can be seen from this figure and also from the Table 2.4 that μ_{PSC} is the square of μ_{RRC}. However, no straight relation seems to exist between μ_{JSC} and μ_{RRC}.

From the Table 2.4, it is found that the μ_{JSC} is always greater than μ_{PSC}, but is smaller as compared to μ_{RRC}. The variability in the values of PSC and JSC are less as compared to that of RRC except for the distributions IX and X which are negatively skewed.

Further, from the Figures 2.1 to 2.10, the following can be observed.

(i) The mode value for PSC is never greater than that for JSC.

(ii) The cumulative distribution curves for CBS and RRC intersect each other at their mean values.

(iii) The cumulative distribution curve for CBS is closest to that of RRC, whereas of JSC is farther and that of PSC farthest.

2.6.3 Value of Commonality Measure Versus Degree of Closeness

As mentioned in (c), the issue of viewing the level of closeness between the two parts of a pair from the numerical values of the commonality measures itself, is quite important when the procedures used for the determination of groups use some threshold value as a cutoff point. In case of extreme situations, i.e. of no matching or complete matching, the commonality measures will also assume extreme values. Therefore, making inference regarding the closeness between the parts for such extreme values is trivial, and vice versa. However, in case when values of the

commonality measures are not at extreme, it is difficult to say exactly as what kind of closeness exists between the two parts of a pair.

In general, the values of any commonality measure will be high only when there is a high degree of matching in requirements of the two parts of a pair. Therefore, a high value of a measure will always correspond to high level of closeness between the two parts of the related pair.

For the pairs of parts, where a part though providing less matching to the machine requirements of the other part, has high matching with that of the other, the values of JSC and PSC will be very low, that of CBS somewhat good, and that of one of the two RRCs quite high and of other quite low. Therefore, in case of JSC and PSC, it will be wrong to say that lower values of the measure will always correspond to low degree of closeness between the two parts of a pair; whereas this statement will hold true for CBS. In case of RRC, the decision can be made only when both of the values of RRC obtained for each of the two parts forming the pair have been looked into. If both of the values are low, it can be concluded that matching between the two parts is poor; and when out of the two at least one is good, the two parts still enjoy high consideration for grouping together.

The discussions made in the above paragraph indicate that the RRC compared to the other measures may involve more number of pairs of parts in grouping decisions. This point can further be studied in the following manner.

—— Keeping the threshold value (say T) for each of the commonality measures (except CBS) to be the same, and then

finding percentage of occurrences of the pairs for which the measure values are above their corresponding threshold values. The threshold value for CBS will be twice of that for the other measures.

— Choosing the threshold values for the various commonality measures in a manner such that the percentage occurrences of the values of the measures beyond their corresponding chosen limits are the same.

2.6.3.1 SAME THRESHOLD VALUE FOR EACH COMMONALITY MEASURE

Most of the heuristic algorithms, which are based on some kind of coefficients representing a commonality measure, require some threshold value to help in deciding the parts which are to be kept in the same group, and in some cases the number of groups as well.

Normally, the threshold values would be kept large to ensure that only similar parts stay in the same group. For higher threshold values, it can be seen from the Figures 2.1 to 2.10 that for comparatively more number of occurrences the value of at least one of the two RRCs computed for a pair of parts is greater than or equal to its specified threshold limit T_{RRC}. For example, for threshold value of 0.7 (1.4 for CBS) for the distribution V, the percentage occurrences beyond the corresponding threshold values for measures RRC, CBS, JSC and PSC are 30, 18, 6 and 5, respectively (see Figure 2.5).

Further, looking into the cumulative frequency diagrams shown in the Figures 2.1 to 2.10, one can infer that if the threshold values are kept sufficiently low such that $T_{RRC} < \mu_{RRC}$ or $T_{CBS} < \mu_{CBS}$, then the measure CBS will find maximum occurrences

beyond its corresponding threshold value. In fact, this does not happen because of the use of two distinct numbers of RRC corresponding to the two parts of a pair compared to only one for the other measures. This aspect is discussed in detail in the following section.

2.6.3.2 DIFFERENT THRESHOLD VALUES FOR COMMONALITY MEASURES

In the previous section, for the purpose of analysis, the same threshold value (in case of CBS, twice of the others) was taken for each of the measures of commonality although the definitions of these measures are different. Therefore, understanding that each measure is different from the other and for the same degree of closeness assumes a different value, it may be reasonable to define different threshold values for different commonality measures.

From the above, it is obvious that in the use of grouping algorithms which involve some threshold value, the distribution of closeness and the measure to be used must be considered for fixing the value of the threshold limit for obtaining the desired (number of) groups.

For the purpose of analyzing which measure involves the maximum pairs of parts for the grouping decisions, threshold values are chosen in such a way that for each of the measures the percentage occurrences beyond the corresponding threshold limit is the same. Such limits can be determined for each possible matching distributions from the cumulative frequency distribution curves shown in the Figures 2.1 to 2.10.

For example, for the matching distribution being normal, the threshold limits (T_{RRC} for RRC_{ij} and RRC_{ji}, T_{CBS} for CBS_{ij}, T_{JSC}

for JSC_{ij} and PSC_{ij}) on the values of the commonality measures for all possible combinations of the parameters of parts i and j forming a pair can be found from their respective cumulative distribution curves shown in the Figure 2.7. The limits for capturing 40% of the total occurrences (i.e. L = 0.4) for which the measure values are beyond their corresponding limits, are given below.

Commonality Measure	Threshold Value
RRC_{ij}	0.541
RRC_{ji}	0.541
CBS_{ij}	1.059
JSC_{ij}	0.268
PSC_{ij}	0.400

Fixing the threshold value in a manner discussed above, the investigations are carried out in the following ways.

(i) Determining as for how many additional pairs of parts, either of the two RRCs for a pair of parts is greater than their corresponding threshold values (T_{RRC}).

(ii) Determining as for how many pairs of parts, the value of at least one of the two RRCs of a pair of parts is not less than the corresponding specified limit (T_{RRC}) and that of a

different measure less than the limit value, and vice versa.

For the above purpose, again a simulation has been carried out considering 1,000,000 pairs of parts each for a different setting of threshold values of commonality measures obtained varying L from 0.1 to 0.9 in step of 0.1.

For the case of RRC, in counting of occurrences beyond the specified limit T_{RRC} those pairs are included for which the values of either of the two RRCs of the parts of a pair are greater than or equal to the specified limit. The results of the simulation are summarized in Figures 2.13 to 2.16.

Figure 2.13 shows the number of additional pairs for RRC over the other commonality measures for which at least one of two RRCs of the parts of a pair is greater than or equal to T_{RRC}. The number of additional pairs encountered by RRC are almost the same for each distribution of matching requirements shown in the Table 2.3, and also over each of the measures CBS, JSC and PSC. It is seen from this figure that the additional occurrences for RRC are maximum at L = 0.5. Similar results can also be found analytically as described below.

Let the threshold limits be decided to capture fraction x of the total occurrences (i.e. L = x) for which the values of the commonality measures are greater than or equal to their respective threshold values. Further, let the fraction of the total occurrences for which either of the two RRCs corresponding to the two parts of a pair is greater than or equal to T_{RRC} be P. Since the two valves of RRCs are independent of each other, P can be expressed as:

$$P = x + x - x.x. \tag{2.41}$$

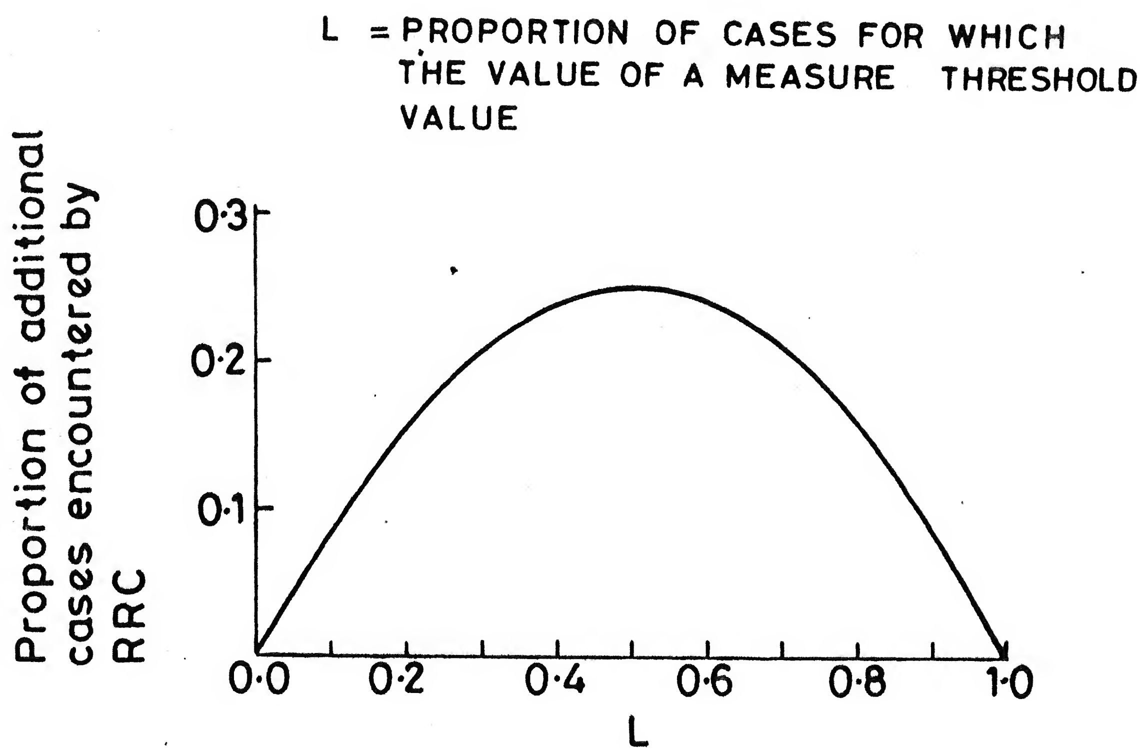


Figure 2·13: Proportion of Additional Pairs for which Atleast one of the RRC for a Pair of Parts is Not Less Than the Threshold Limit ($=T_{RRC}$).

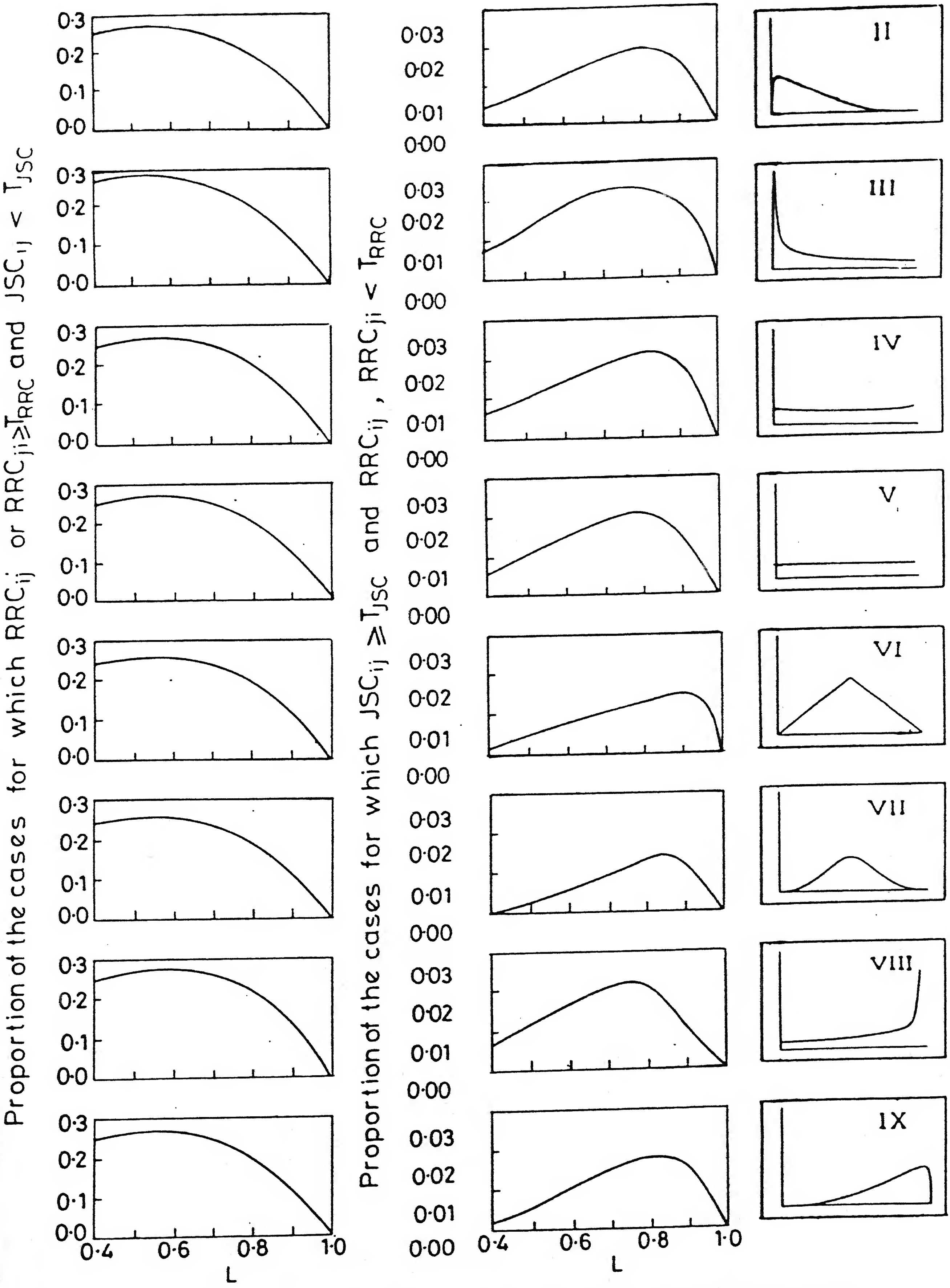


Figure 2·15: Proportion of the Cases Missed by One Acknowledged by the Other Measure (Measures: RRC and JSC).

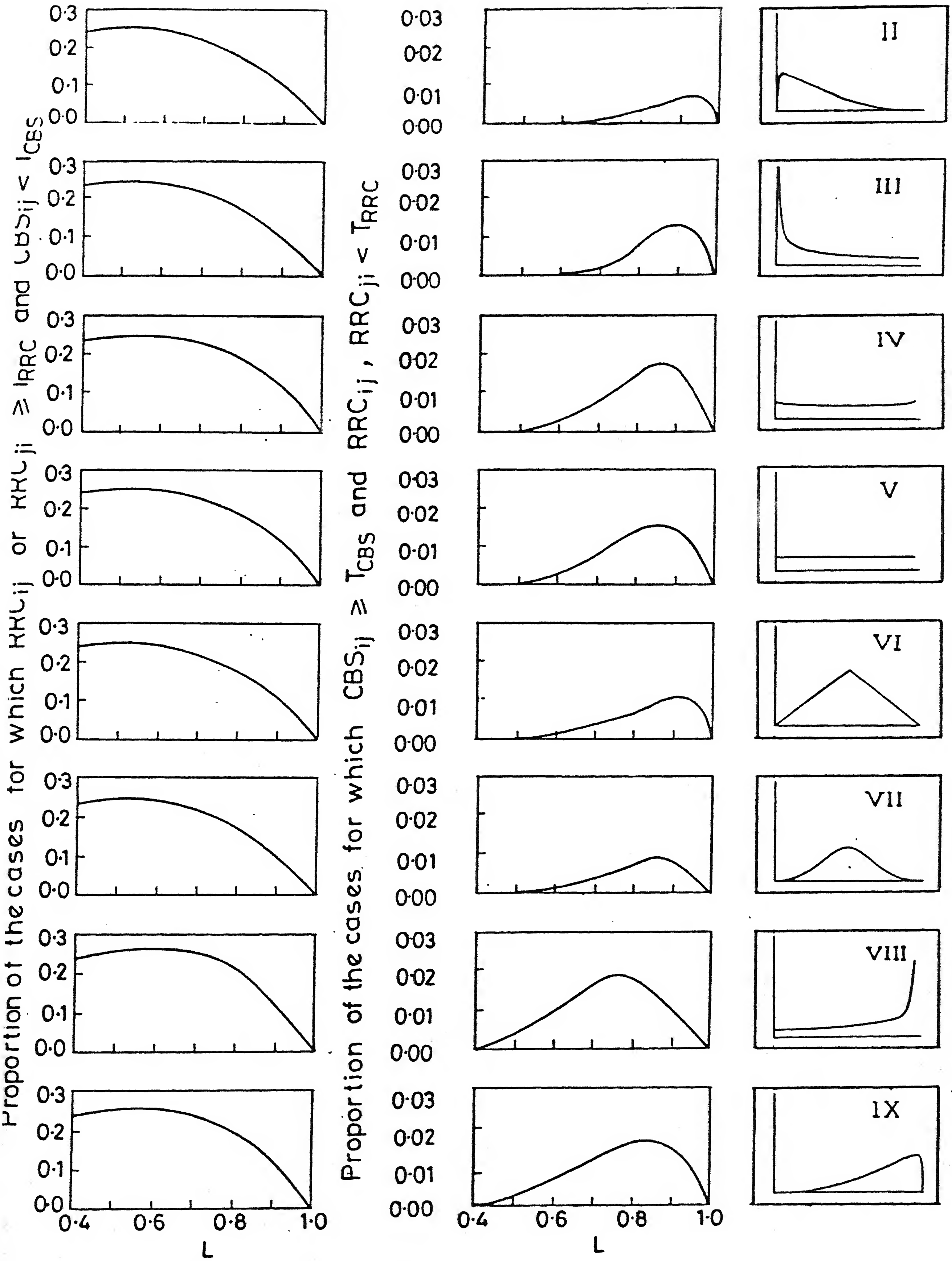


Figure 2·14:Proportion of the Cases Missed by One and Acknowledged by the Other Measure (Measures : RRC and CBS).

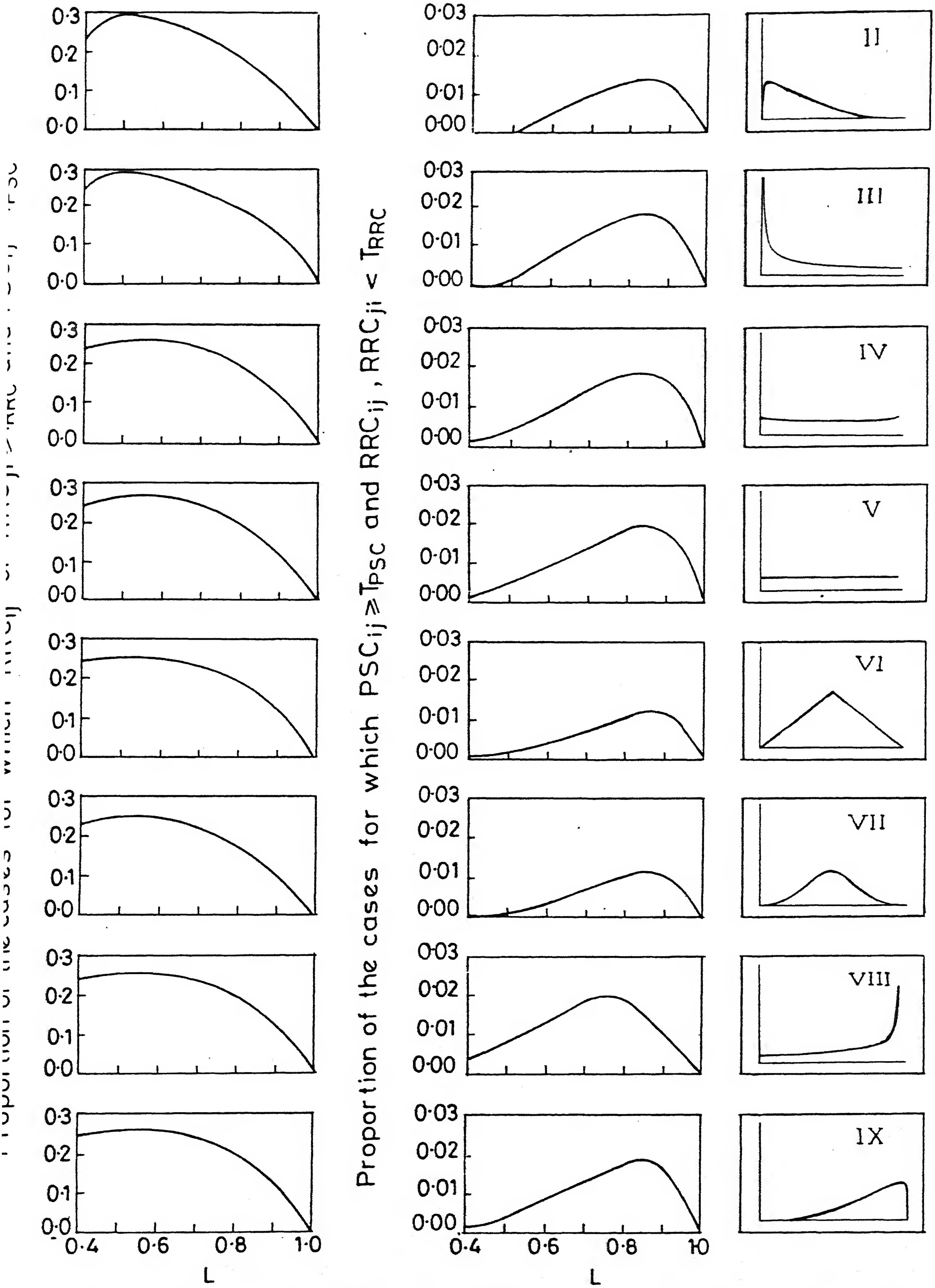


Figure 2·16: Proportion of the Cases Missed by One and Acknowledged by the Other Measure (Measures: RRC and PSC).

Therefore, the fraction of the additional cases encountered by RRC, denoted as D, will be:

$$D = P - x.$$

In view of the equation (2.41), the above equation can be written as,

$$D = x - x^2. \tag{2.42}$$

Since x is at most equal to one, $(x-x^2)$ will never be negative, and thus D will always be greater than or equal to zero. It implies that RRC as compared to the other measures will definitely encounter more cases.

Further, on setting $\frac{dD}{dx} = 0$, x is found equal to 0.5. Since $\frac{d^2D}{dx^2}$ is less than zero at x = 0.5, D will be maximum at x = 0.5. That is, the number of additional cases for RRC will be maximum when the threshold values are set at L = 0.5. From the equation (2.42), it is obvious that D will be equal to zero if x is equal to zero or one.

Further from the Figures 2.14, 2.15 and 2.16, it can be seen that the number of cases which are noticed by RRC but not by the other measures beyond their respective threshold values, are almost the same for each of the measures CBS, JSC and PSC, and this characteristic is the same for all the ten distributions considered for requirement matching. The number of such occurrences are maximum for the value of L equal to and around 0.5. The number of such occurrences increases with the increase in L upto L = 0.5, and beyond this the number starts decreasing.

It can further be observed that for the lower values of L, the number of occurrences encountered by the other measures but not by RRC is almost negligible. The number increases with the increase in value of L upto L = 0.8 and around.

As discussed earlier, threshold values are normally kept somewhat large for the purpose of grouping. Since the higher values of threshold correspond to the lower values of L, it can be inferred from the observations made in the above paragraphs that beyond the respective threshold values RRC will capture large number of cases as what cannot be noticed by others, whereas the other measures will not observe such occurrences or at the best these will be negligibly small.

2.7 NUMERICAL EXAMPLES

To show the use of RRC and also its superiority over the measures ARC, CBS, JSC and PSC, some illustrative numerical examples are considered. For a valid and true comparison, the algorithm or the grouping methodology chosen should be having characteristic of adopting any of these commonality measures. The p-median formulation of the grouping problem proposed by Kusiak (1987) serves this purpose. The formulation of the grouping problem for the parts having only single process plan is given below.

p-median Formulation:

$$\text{Maximize} \quad \sum_{i=1}^{n} \sum_{j=1}^{n} s_{i,j}\, x_{i,j}$$

Subject to:

$$\sum_{j=1}^{N} x_{i,j} = 1 \qquad i = 1, \ldots, N$$

$$\sum_{j=1}^{N} x_{j,j} = p$$

$$x_{i,j} \leq x_{j,j} \qquad i, j = 1, \ldots, N$$

$$x_{i,j} = 0 \text{ or } 1 \qquad i, j = 1, \ldots, N$$

where

N = total number of parts

p = number of required groups

$s_{i,j}$ = commonality measure value for a pair of parts i and j (of part i with respect to part j in case of ARC and RRC)

$$x_{i,j} = \begin{cases} 1 & \text{if part } i \text{ is assigned to part family } j \\ 0 & \text{otherwise} \end{cases}$$

Each of the examples shown in Figures 2.17 to 2.21 are formulated as p-median problem described above and are solved on DEC-10 computer using mathematical programming software LINDO. The solutions obtained are listed in Tables 2.5 to 2.11. In the incidence matrices of the examples, 'X' represents requirement of a machine by the corresponding part.

Machine \ Part	1	2	3	4	5
1	X	X		X	
2	X			X	
3	X	X		X	
4		X	X		X
5	X			X	
6	X	X		X	
7		X	X		
8	X	X		X	
9					X
10		X	X		X
11	X	X		X	
12	X	X		X	
13	X	X		X	
14	X			X	
15		X	X		X
16	X			X	
17	X			X	
18		X	X		X

Figure 2.17: Example 2.1 (Seifoddini (1989c), original matrix is transposed).

Table 2.5: Solution to Example 2.1 for p = 2.

Commonality measure	Part families	Number of exceptional elements	Total machines required	CPU time (sec)
ARC	{1, 4} and {2, 3, 5}	7	25	1.33
CBS	{1, 4} and {2, 3, 5}	7	25	1.79
JSC	{1, 4} and {2, 3, 5}	7	25	1.72
PSC	{1, 4} and {2, 3, 5}	7	25	1.70
RRC	{1, 4} and {2, 3, 5}	7	25	1.35

Machine \ Part	1	2	3	4	5	6	7	8	9
1			X						X
2					X				
3	X	X							
4	X				X				
5				X			X	X	
6				X		X			
7			X						X
8	X	X			X				
9	X	X			X				
10	X				X				
11				X			X	X	
12		X							
13	X	X			X				
14						X			
15	X				X				

Figure 2.18: Example 2.2 (Kumar et al (1986)).

Table 2.6: Solution to Example 2.2 for p = 2.

Commonality measure	Part families	Number of exceptional elements	Total machines required	CPU time (sec)
ARC	{1, 2, 3, 5, 9} and {4, 6, 7, 8}	0	15	4.39
CBS	{1, 2, 3, 5, 9} and {4, 6, 7, 8}	0	15	3.44
JSC	{1, 2, 3, 5, 6, 9} and {4, 7, 8}	1	16	3.80
PSC	{1, 2, 3, 5, 6, 9} and {4, 7, 8}	1	16	3.89
RRC	{1, 2, 3, 5, 9} and {4, 6, 7, 8}	0	15	3.95

Part

Machine	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1						X	X	X		X						
2		X				X		X	X					X		X
3											X		X			
4									X							
5				X	X										X	
6						X								X		
7						X										X
8					X	X		X								
9				X	X			X								
10		X							X							X
11												X				
12						X		X		X						
13						X	X			X						
14				X	X	X									X	
15					X			X								
16					X											
17						X								X		
18									X							X
19				X	X	X		X							X	
20											X					
21				X	X			X							X	
22												X				
23				X	X	X		X								
24											X	X	X			
25							X			X						
26										X						
27											X					
28		X						X	X							
29				X	X											
30											X	X				
31								X		X						
32		X				X			X							X
33					X	X									X	
34			X													
35														X		
36			X													
37	X	X				X		X	X							X
38		X						X	X							X
39						X				X						
40		X				X			X							
41					X			X							X	
42	X	X				X			X							X
43					X	X		X							X	

Figure 2.19: Example 2.3 (King and Nakornchai (1982)).

Table 2.7: Solution to Example 2.2 for p = 3.

Commonality measure	Part families	Number of exceptional elements	Total machines required	CPU time (sec)
ARC	{1, 2, 5}, {3, 9} and {4, 6, 7, 8}	0	15	3.84
CBS	{1, 2, 5}, {3, 9} and {4, 6, 7, 8}	0	15	3.34
JSC	{1, 2, 5}, {3, 6, 9} and {4, 7, 8}	1	16	3.34
PSC	{1, 2, 5}, {3, 6, 9} and {4, 7, 8}	1	16	3.36
RRC	{1, 2, 5}, {3, 9} and {4, 6, 7, 8}	0	15	3.42

Table 2.8: Solution to Example 2.3 for p = 2.

Commonality measure	Part families	Number of exceptional elements	Total machines required	CPU time (sec)
ARC	{1, 2, 3, 6, 7, 8, 9, 10, 11, 14, 16} and {4, 5, 12, 13, 15}	16	56	18.13
CBS	{1, 2, 4, 5, 6, 7, 8, 9, 10, 14, 15, 16} and {3, 11, 12, 13}	0	43	20.95
JSC	{1, 2, 7, 9, 10, 11, 13, 14, 16} and {3, 4, 5, 6, 8, 12, 15}	20	60	24.19
PSC	{1, 2, 3, 6, 7, 9, 10, 11, 12, 13, 14, 16} and {4, 5, 8, 15}	13	56	27.56
RRC	{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 14, 15, 16} and {11, 12, 13}	0	43	35.06

Machine	Part 1	2	3	4	5	6	7	8
1					X	X		
2	X		X					
3	X	X		X			X	X
4		X		X			X	X
5					X	X		
6		X		X	X		X	X
7		X		X			X	X
8	X		X					
9	X		X			X		
10				X	X	X		
11	X		X				X	
12					X	X	X	
13	X		X					
14	X	X	X					
15					X	X		
16	X		X					
17	X		X		X			
18		X		X			X	X
19	X		X					
20		X		X		X	X	X

Figure 2.20: Example 2.4 (Chandrashekharan and Rajagopalan (1986): original matrix is transposed).

Table 2.9: Solution to Example 2.4 for p = 2.

Commonality measure	Part families	Number of exceptional elements	Total machines required	CPU time (sec)
ARC	{1, 3, 6} and {2, 4, 5, 7, 8}	10	30	2.92
CBS	{2, 4, 5, 6, 7, 8} and {1, 3}	5	25	5.33
JSC	{2, 4, 5, 6, 7, 8} and {1, 3}	5	25	6.48
PSC	{2, 4, 5, 6, 7, 8} and {1, 3}	5	25	6.55
RRC	{2, 4, 5, 6, 7, 8} and {1, 3}	5	25	3.36

Machine	Part 1	2	3	4	5	6	7	8	9
1	X	X		X	X				
2	X	X							
3	X	X		X	X				
4	X	X							
5	X				X				
6	X	X		X					
7	X				X				
8	X			X					
9	X		X						
10	X		X						
11	X		X	X					X
12			X			X	X		X
13			X			X	X	X	X
14						X			
15						X	X	X	

Figure 2.21: Example 2.5.

Table 2.10: Solution to Example 2.5 for p = 2.

Commonality measure	Part families	Number of exceptional elements	Total machines required	CPU time (sec)
ARC	{1, 2, 3, 4, 5} and {6, 7, 8, 9}	3	18	4.25
CBS	{1, 2, 4, 5} and {3, 6, 7, 8, 9}	4	18	3.68
JSC	{1, 2, 4, 5} and {3, 6, 7, 8, 9}	4	18	3.69
PSC	{1, 2, 4, 5} and {3, 6, 7, 8, 9}	4	18	5.83
RRC	{1, 2, 3, 4, 5} and {6, 7, 8, 9}	3	18	3.90

Table 2.11: Solution to Example 2.5 for p = 3.

Commonality measure	Part families	Number of exceptional elements	Total machines required	CPU time (sec)
ARC	{1, 2, 4, 5}, {3, 9} and {6, 7, 8}	8	20	3.41
CBS	{1, 2, 4, 5}, {3, 9} and {6, 7, 8}	8	20	3.99
JSC	{1, 2, 4, 5}, {3, 9} and {6, 7, 8}	8	20	4.64
PSC	{1, 2, 4, 5}, {3, 9} and {6, 7, 8}	8	20	4.17
RRC	{1, 2, 4, 5}, {3, 9} and {6, 7, 8}	8	20	3.92

Table 2.12: Solution to Example 2.5 for p = 4.

Commonality measure	Part families	Number of exceptional elements	Total machines required	CPU time (sec)
ARC	{1, 2, 4}, {3, 9}, {5} and {6, 7, 8}	10	22	3.57
CBS	{1}, {2, 4, 5}, {3, 9} and {6, 7, 8}	12	25	4.24
JSC	{1}, {2, 4, 5}, {3, 9} and {6, 7, 8}	12	25	7.11
PSC	{1}, {2, 4, 5}, {3, 9} and {6, 7, 8}	12	25	4.12
RRC	{1, 2, 4, 5}, {3, 9}, {6, 7} and {8}	10	22	4.48

In the Tables 2.5 to 2.12, 'total machines required' represents the total number of machines needed for a completely disjoint cellular manufacturing, i.e. when all the required machines are assigned to each group of parts. However, for the purpose of computation of exceptional elements it is assumed that only one copy of each machine type is available and the machines are assigned to those part families that require them most.

The solutions presented in the Tables 2.5 to 2.11 show that for RRC the requirements of duplicate machines and the number of exceptional elements are less than or equal to that for the other commonality measures. It can be seen from the Tables 2.5 and 2.11 that the CPU time requirement for ARC as compared to that for the other measures is less when all result the same solution. It can also be observed that in case of CBS the requirements of machines and the number of exceptional elements are at most equal to that when the JSC or the PSC are used.

Now the result obtained for Example 5 where two groups are to be formed, is analyzed to see as how the non-symmetric property of RRC compared to the symmetric property of the other measures helps in producing better results. It can be seen from the solution given in the Table 2.10 that the number of exceptional elements becomes 3 instead of 4 if part 3 is assigned to the group containing parts 1, 2, 4 and 5 instead of to the group of parts 6, 7, 8 and 9. Removing part 3 from consideration though reduces the number of exceptional elements but does not make the two groups completely disjoint. However, when part 9 is also taken out from grouping consideration the disjoint groups of parts and machines

are as given below.

Part families : {1, 2, 4, 5} and {6, 7, 8}.

Machine cells : {1, 2, 3, 4, 5, 6, 7, 8, 10, 11} and {14, 13, 14, 15}.

It can be seen from the incidence matrix shown in Figure 2.21 and the solution given above that the parts 1 and 6 are the two dominant parts in the two respective groups with whom the machine requirements of the parts belonging to the same group are having complete matching. Now the parts 3 and 9 are to be assigned to the two groups such that the total number of duplicate machines and also of exceptional elements are minimized.

From the relative requirement compatibility matrix shown in Figure 2.22, it can be observed that the RRC of the part 3 with respect to some other part is maximum for the part 1 ($RRC_{3,1}$ = 0.600). And, $RRC_{9,1}$ and $RRC_{9,4}$ are low as compared to $RRC_{9,6}$. Therefore, part 3 will stay with part 1 and part 9 with part 6. Such assignments are already shown to require the least number of additional machines and to have the minimum number of exceptional elements.

j

i	1	2	3	4	5	6	7	8	9
1	1.000	0.455	0.273	0.455	0.364	0.000	0.000	0.000	0.091
2	1.000	1.000	0.000	0.600	0.400	0.000	0.000	0.000	0.000
3	0.600	0.000	1.000	0.200	0.000	0.400	0.400	0.200	0.600
4	1.000	0.600	0.200	1.000	0.400	0.000	0.000	0.000	0.200
5	1.000	0.500	0.000	0.500	1.000	0.000	0.000	0.000	0.000
6	0.000	0.000	0.500	0.000	0.000	1.000	0.750	0.500	0.500
7	0.000	0.000	0.667	0.000	0.000	1.000	1.000	0.667	0.667
8	0.000	0.000	0.500	0.000	0.000	1.000	1.000	1.000	0.500
9	0.333	0.000	1.000	0.333	0.000	0.667	0.667	0.333	1.000

Figure 2.22. Relative Requirement Compatibility Matrix $[RRC_{ij}]$ for Example 2.5.

j

		1	2	3	4	5	6	7	8	9
	1	0.000	1.455	0.873	1.455	1.364	0.000	0.000	0.000	0.424
	2	1.455	0.000	0.000	1.200	0.900	0.000	0.000	0.000	0.000
	3	0.873	0.000	0.000	0.400	0.000	0.900	1.067	0.700	1.600
	4	1.455	1.200	0.400	0.000	0.900	0.000	0.000	0.000	0.533
i	5	1.364	0.900	0.000	0.900	0.000	0.000	0.000	0.000	0.000
	6	0.000	0.000	0.900	0.000	0.000	0.000	1.750	1.500	1.167
	7	0.000	0.000	1.067	0.000	0.000	1.750	0.000	1.667	1.333
	8	0.000	0.000	0.700	0.000	0.000	1.500	1.667	0.000	0.833
	9	0.424	0.000	1.600	0.533	0.000	1.167	1.333	0.833	0.000

Figure 2.23. Cell Bond Strength Matrix $[CBS_{ij}]$ for Example 2.5.

j

		1	2	3	4	5	6	7	8	9
	1	0.000	0.455	0.164	0.455	0.364	0.000	0.000	0.000	0.030
	2	0.455	0.000	0.000	0.360	0.200	0.000	0.000	0.000	0.000
	3	0.164	0.000	0.000	0.040	0.000	0.200	0.267	0.100	0.600
	4	0.455	0.360	0.040	0.000	0.200	0.000	0.000	0.000	0.067
i	5	0.364	0.200	0.000	0.200	0.000	0.000	0.000	0.000	0.000
	6	0.000	0.000	0.200	0.000	0.000	0.000	0.750	0.500	0.333
	7	0.000	0.000	0.267	0.000	0.000	0.750	0.000	0.667	0.444
	8	0.000	0.000	0.100	0.000	0.000	0.500	0.667	0.000	0.167
	9	0.030	0.000	0.600	0.067	0.000	0.333	0.444	0.167	0.000

Figure 2.24. Product Type Similarity Coefficient Matrix $[PSC_{ij}]$ for Example 2.5.

j

		1	2	3	4	5	6	7	8	9
	1	0.000	0.455	0.231	0.455	0.364	0.000	0.000	0.000	0.077
	2	0.455	0.000	0.000	0.429	0.286	0.000	0.000	0.000	0.000
	3	0.231	0.000	0.000	0.111	0.000	0.286	0.333	0.167	0.600
	4	0.455	0.429	0.111	0.000	0.286	0.000	0.000	0.000	0.143
i	5	0.364	0.286	0.000	0.286	0.000	0.000	0.000	0.000	0.000
	6	0.000	0.000	0.286	0.000	0.000	0.000	0.750	0.500	0.400
	7	0.000	0.000	0.333	0.000	0.000	0.750	0.000	0.667	0.500
	8	0.000	0.000	0.167	0.000	0.000	0.500	0.667	0.000	0.250
	9	0.077	0.000	0.600	0.143	0.000	0.400	0.500	0.250	0.000

Figure 2.25. Jaccard's Similarity Coefficient Matrix $[JSC_{ij}]$ for Example 2.5.

Further, from the values of CBS, JSC, and PSC given in the Figures 2.23, 2.24 and 2.25, it can be noticed that the parts 3 and 9 are closest to part 1 in the first group (having parts 1, 2, 4, and 5) and that with part 7 in the second group (having parts 6, 7 and 8). In addition, the values of these measures showing the amount of closeness between parts 3 and 7 are higher as compared to their respective values for the pair of part 3 and 1. The same holds true for the pair of parts 9 and 7 over the other pair of parts 9 and 1. Thus, for the objective of maximizing the total similarity, parts 3 and 9 should be assigned to the second group. It is this reason because of which the use of the other measures assigns parts 3 and 9 to the group of parts 6, 7 and 8, and requires more number of duplicate machines for perfect grouping.

In the preceding paragraph, it has been seen as how the value of $RRC_{3,1}$ helps regarding the assignment of the part 3 to the group of part 1. Next, it is discussed as how the other value of RRC, i.e. $RRC_{1,3}$ decides grouping.

Let the two part families be {2, 4, 5} and {3, 6, 7, 8, 9}. Since RRC_{13} is less as compared to RRC_{12} or RRC_{14}, the part 1 instead of being assigned to the group of part 3 will be placed in the group of parts 2 and 4. Now the part 1 will attract part 3 for placement in the same group along with parts 2, 4 and 5 because RRC of part 3 is maximum with respect to part 1. This finally results the best possible grouping.

2.8 SUMMARY AND CONCLUSIONS

This chapter presents two new measures of commonality for GT applications for general and flexible production situations, such

as FMS, where parts can have alternative process plans, and two or more operations in a process plan may require machines of the same type and capability. The two measures, termed as Relative Requirement Compatibility (RRC) and Absolute Requirement Compatibility (ARC), are defined on the bases of processing times, the number of the operations, and the number of the machine and\or tool types required for processing the operations in various process plans.

The mathematical relation of the relative requirement compatibility with other commonality measures, such as, Jaccard's similarity coefficient (JSC), product type similarity coefficient (PSC) and cell bond strength (CBS), are established. The relative requirement compatibility, based on these relations, is reported to possess better discriminating characteristic. The illustrations presented confirm this observation and show the better discriminating characteristic of the relative requirement compatibility compared to that of the simple matching similarity coefficient. Several numerical examples solved using p-median approach of Kusiak (1987) show that the relative requirement compatibility does provide better results.

An analysis performed on the values of these commonality measures shows their comparative discriminating power and provides an insight for the determination of the threshold value for each of the commonality measures which are needed by certain grouping algorithms and methodologies.

For a generalized case when there is a limit on the total number of groups and also on the number of parts and machines which can be assigned to a group, the consideration of the

absolute requirement compatibility along with relative requirement compatibility can be of great help.

CHAPTER III

GENERALIZED GROUPING PROBLEMS: BASIC PROBLEM AS A p-MEDIAN PROBLEM

3.1 INTRODUCTION

The advantages of GT concepts have been attributed to the decisions both at the system design level as well as at the operations level. At the design level decisions are taken for the determination of the number and types of machines and their layout, selection of material handling system, etc., and at the operational for loading, scheduling etc..

The present chapter focuses on the grouping problem related to the effective utilization of the machines in the context of cellular manufacturing system. The effective machine utilization is achieved by classifying parts and machines into various groups to yield maximum total similarity. In addition, the problem of determining the number of machines of each type to be procured at a minimum investment is also considered.

Certain factors related to the system and its configuration, such as, the total number of groups, the total number of machines of each type and their capacities, and the limits on the number of machines and parts in a group are considered to act as the constraints for these problems. These constraints, in general, tend to drive the grouping solution away from the perfect group formation in which the groups are disjoint and parts do not have to move to any other group. The succeeding paragraphs present some discussions on these constraints.

The constraint on the total number of groups to be formed is important not only from the operational viewpoint, but also from the organizational perspective. The number of groups influences intra and intercell movements, work load distribution on the machines and the size of the groups. A small number of groups would usually result in higher intracell movement but lower intercell movement, and into relatively large group size. A large number of groups would, obviously, have opposite effects.

The consideration of the limit on the total number of machines becomes an important factor while designing a new production system or appending some additional machines into existing one from the viewpoint of total investment, space requirement, material handing system (e.g., AGVs, tow carts, conveyors, etc.) and their layout.

Provision of more than one copy of the same machine and their proper distribution among groups makes the group of machines (called as machine cells) organizationally and operationally more independent, and reduces intercell movements. The consideration of the constraint on number of machines of each type, therefore, would help in judicious assignment of the machines to the cells.

The constraint, on the maximum number of machines which can be assigned to a group, prevents the machine cells to grow beyond a reasonable limit. Assigning too many machines to a cell may not result the desired benefits of GT and even the intracell movement may become significant. On the other hand, machine cells with very few machines would involve enormous intercell movements and thus may not be economical, and at the same time may lead to certain organizational problems. Thus, for effective resource

utilization, both upper and lower limits on the number of machines in a cell need to be considered. For the similar reasons, the consideration of limits on the minimum and the maximum number of parts that can be assigned to a group becomes necessary. A large number of parts when assigned to a group may cause route congestions and too few may result into under utilization of the resources.

The machine capacity, in general, is not so relevant to the problem of grouping based on the technological relationship of parts and machines, but helps in determining the number of machines of each type and thus their total number. However, its consideration becomes important for those grouping situations where parts have alternative process plans or when the production system has versatile machines such as in Flexible Manufacturing Systems (FMS).

In the traditional simple grouping problem, an operation of a part can be performed on only one machine type. For these grouping problems normally the situations where the two parts though not quite close to each other in terms of their total machine requirements but requiring some common machines, restrict the formation of perfect groups and result intercell movements. Of course, improvements towards perfect grouping are possible by merging together the related groups or by assigning such common machines to each concerned group.

In the manufacturing systems, such as FMS, where an operation of a part may be performed on more than one machine type, or where a machine can perform more than one operation type, the parts will have more than one process plan. For such cases, it is possible

to obtain solution closer to perfect grouping as compared to the case of parts having single process plan. This is so because most of the requirements of a part can be satisfied in the same group (i) due to versatility of the machines and also (ii) by selecting some suitable process plan out of the several alternatives. The objective of obtaining perfect grouping may, however, be impeded by the capacity of the machines. Nevertheless, the machine capacity consideration leads to the determination of process plans (each for a different part) in a manner such that the processing load on each machine does not exceed the available capacity. Further, the chosen process plans can be used to directly identify the machines required for processing the operations of the various parts. Thus, the generalized grouping problems with capacity consideration would provide solutions feasible even to the machine loading problems.

It may be noted that the machine capacity constraints play no role in forming the groups in case of the simple grouping.

From the above, it is evident that the input required for the simple grouping consists of a part machine matrix showing the requirement of machines by parts, whereas in case of the generalized grouping, additional information on processing time requirement and the machine capacity are also needed.

The solution methodologies available for simple grouping problem can broadly be categorized in two classes: one in which the groups of parts and machines are determined hierarchically, and the other in which the groups are formed simultaneously. The procedures developed for these methodologies have been either optimization, or some kind of heuristics exploiting certain

aspects of the problem. Details of these approaches are given by King and Nakornchai (1982), Waghodekar and Sahu (1984), and Ballakur and Studel (1987).

The generalized grouping problem has attracted relatively less attention from the researchers. Kusiak (1987) has formulated the problem as a modified p-median problem. The problem has been modeled as a generalized assignment problem by Shtub (1989). In both of the cases, though the approaches are different, solution obtained may remain the same.

In the present chapter, the formulations proposed by Kusiak (1987) and Shtub (1989) have been reviewed and improved in terms of the number of constraints and variables. Improvements to the Kusiak's model reduce the required number of constraints, whereas in case of Shtub's model, improvements reduce both the required number of variables and constraints. In addition, the generalized grouping problem has been considered with several extensions and generalizations over the previous models incorporating versatility and capacity of the machines, flexibility for the operations and disjointedness of groups.

In the objective function of the proposed models, the relative requirement compatibility (RRC) is incorporated as a measure of commonality instead of the measure used by Kusiak and Shtub. The measure RRC as discussed in the Chapter II is more generalized and is shown to provide better grouping solutions.

In Section 3.2, formulations proposed by Kusiak (1987) and Shtub (1989) are presented, and the related observations regarding the grouping approach, the formulation and the input are described in Section 3.3. Based on these observations, improvements are

brought in the two formulations and are given in Section 3.4. Section 3.5 contains the extensions of the grouping problem with their mathematical models. In Section 3.6, a heuristic approach for solving some grouping problems is presented. In Section 3.7, the models developed in Section 3.5 are illustrated with the help of numerical examples and the sensitivity of various constraints on the grouping solution is studied. Conclusions and summary of the work are presented in Section 3.8.

3.2 THE FORMULATIONS OF KUSIAK AND SHTUB

The grouping problem formulated by Kusiak (1987) and Shtub (1989) takes into consideration parts with the alternative process plans. The input, providing details of the problem, is a two dimensional (0-1) matrix showing the requirement of machines by various process plans. The objective is to form groups of the process plans such that the total similarity defined in some suitable manner is maximized, and only one process plans from the several alternatives of a part is selected in the group formation. These formulations require the desired number of process families to be prespecified.

The formulations of the problem given by Kusiak (1987) and Shtub (1989) are discussed first, and certain observations on the formulation and the problem structure are made to provide motivations for suitable modifications as well as for developing some models incorporating additional features of the grouping problem.

For presentation of the formulation, the following notations and definitions are used.

Indices

i,j : process plans

n : part

$\underline{m}$: machine

Parameters

$\underline{M}$ = the total number of machines

N = the total number of parts

p = the total number of required process families

q = the total number of process plans over all parts

F_n = set of process plans for part n

$$a_{i,\underline{m}} = \begin{cases} 1 & \text{if process plan } i \text{ requires machine } \underline{m} \\ 0 & \text{otherwise} \end{cases}$$

Derived Parameter

s_{ij} = similarity between process plans i and j

The value of s_{ij} represents the sum of the number of machines common to plans i and j and the number of machines not required by both of the plans. The elements of the similarity coefficient matrix $[s_{ij}]$ are computed as follows.

(i) <u>Similarity between two process plans of different parts</u>

$$s_{ij} = \sum_{\underline{m}=1}^{\underline{M}} \delta(a_{i,\underline{m}}\, a_{j,\underline{m}}) \qquad \forall\ i \in F_{n_1};\ \forall\ j \in F_{n_2};\ n_1 \neq n_2, \text{ and } n_1, n_2 = 1, \ldots, N \tag{3.1}$$

(ii) <u>Similarity between two process plans of the same part</u>

$$s_{ij} = -\infty \qquad i \neq j;\ i,j \in F_n \text{ and } n = 1, \ldots, N \tag{3.2}$$

(iii) <u>Similarity of a process plan with itself</u>

$$s_{ij} = 0 \qquad i = j;\ i,j \in F_n \text{ and } n = 1, \ldots, N \tag{3.3}$$

Metric

$$\delta(c, d) = \begin{cases} 1 & \text{if } c = d \\ 0 & \text{otherwise} \end{cases}$$

Decision Variables

A process family is identified by the index of one of its member process plans. The p required process families, therefore, will assume their indices from the range 1 to q.

For i, j = 1, ..., q, the following decision variable is defined:

$$x_{ij} = \begin{cases} 1 & \text{if process plan } i \text{ belongs to process family } j \\ 0 & \text{otherwise} \end{cases}$$

It may be noted that for the formation of p number of process families, the relation, $p \le N \le q$, must hold.

3.2.1 Kusiak's Formulation

The generalized grouping problem formulated by Kusiak (1987) as a 0-1 integer programming problem is presented briefly. The model is denoted as M3.1 for further references and discussions.

<u>MODEL M3.1</u>

OBJECTIVE FUNCTION:

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} s_{ij}\, x_{ij} \tag{3.4}$$

CONSTRAINTS:

(i) Selection of only one process plan for a part

$$\sum_{i \in F_n} \sum_{j=1}^{q} x_{ij} = 1 \qquad n = 1, .., N \qquad (3.5)$$

(ii) Formation of required number of groups

$$\sum_{j=1}^{q} x_{jj} = p \qquad (3.6)$$

(iii) Assignment of process plans to groups

$$x_{ij} \leq x_{jj} \qquad i = 1,...,q; \quad j = 1,...,q \qquad (3.7)$$

(iv) Decision variables

$$x_{ij} = 0 \text{ or } 1 \qquad i, j = 1,...,q \qquad (3.8)$$

As mentioned earlier, a process family is identified by the index of one of its member process plans, and thus a process family j must include the process plan j, i.e. $x_{jj} = 1$. Further, for the requirement of forming p process families, the expression $\sum_{j=1}^{q} x_{jj}$ must also be equal to p. Therefore, from the definition of x_{ij} and the structure of the objective function, the similarity coefficient matrix $[s_{ij}]$ can be viewed as an assignment matrix along the diagonal of which p assignments must take place (i.e. $\sum_{j=1}^{q} x_{jj} = p$). The rows (with index i) represent the candidate process plans, while the columns (with index j) represent the process families.

3.2.2 Shtub's Formulation

The grouping problem has been formulated as a generalized assignment problem (GAP) by Shtub (1989). The structure of GAP relevant to the present problem as described by him, is as follows:

$$\text{Minimize} \quad \sum_{i \in I} \sum_{j \in J} c_{ij}\, x_{ij}$$

Subject to

$$a_j \le \sum_{i \in I} \gamma_{ij}\, x_{ij} \le b_j \qquad \forall\, j \in J$$

$$\sum_{j \in J} x_{ij} = 1 \qquad \forall\, i \in I$$

where x_{ij} is the decision variable assuming the value equal to one if task i is assigned to agent j, and zero otherwise. Such an assignment is associated with cost c_{ij} and requires resource of amount γ_{ij}. The total cost of assignment is to be minimized while ensuring that the total resource consumption of agent j is within the given limits of a_j and b_j, and a task is assigned to only one agent.

The formulation of GAP has been represented by Shtub in an equivalent form of a matrix as shown in Figure 3.1, and that of the generalized grouping using the basic structure of GAP as shown in Figure 3.2. The two entries, one at the left upper corner and the other at the right bottom, appearing in each of the cell of the matrix represent the values for similarity and the resource requirement, respectively. In this matrix, the columns are taken to represent the task (process plan) and the rows as agents

(process family). The problem can now be viewed as the assignment of process plans to process families such that the total of negative similarity is minimized.

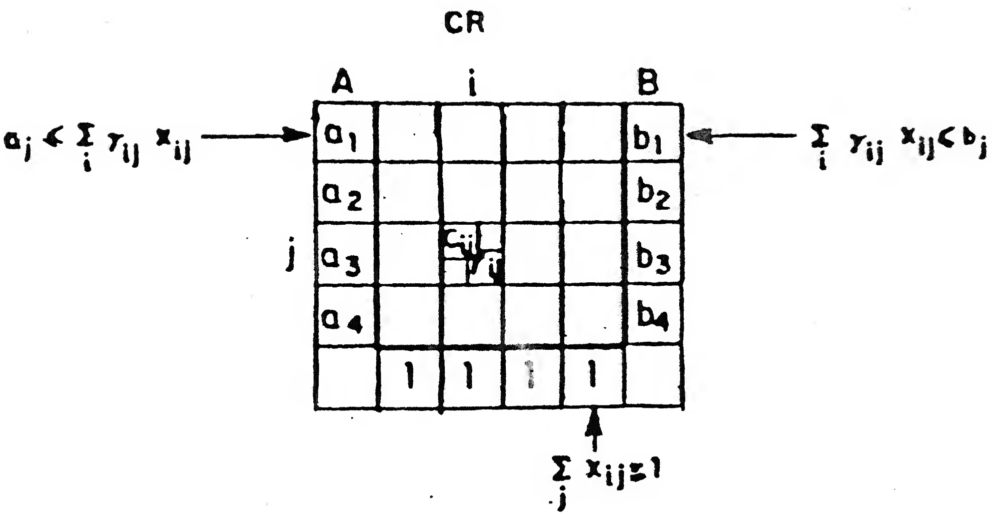


Figure 3.1: The Table Representation of GAP.

The mathematical formulation corresponding to the matrix representation of GAP, as shown in Figure 3.2, requires the following additional 0-1 variables:

$$u_n = \begin{cases} 0 & \text{if some process plan of part n represents a process family} \\ 1 & \text{otherwise} \end{cases}$$

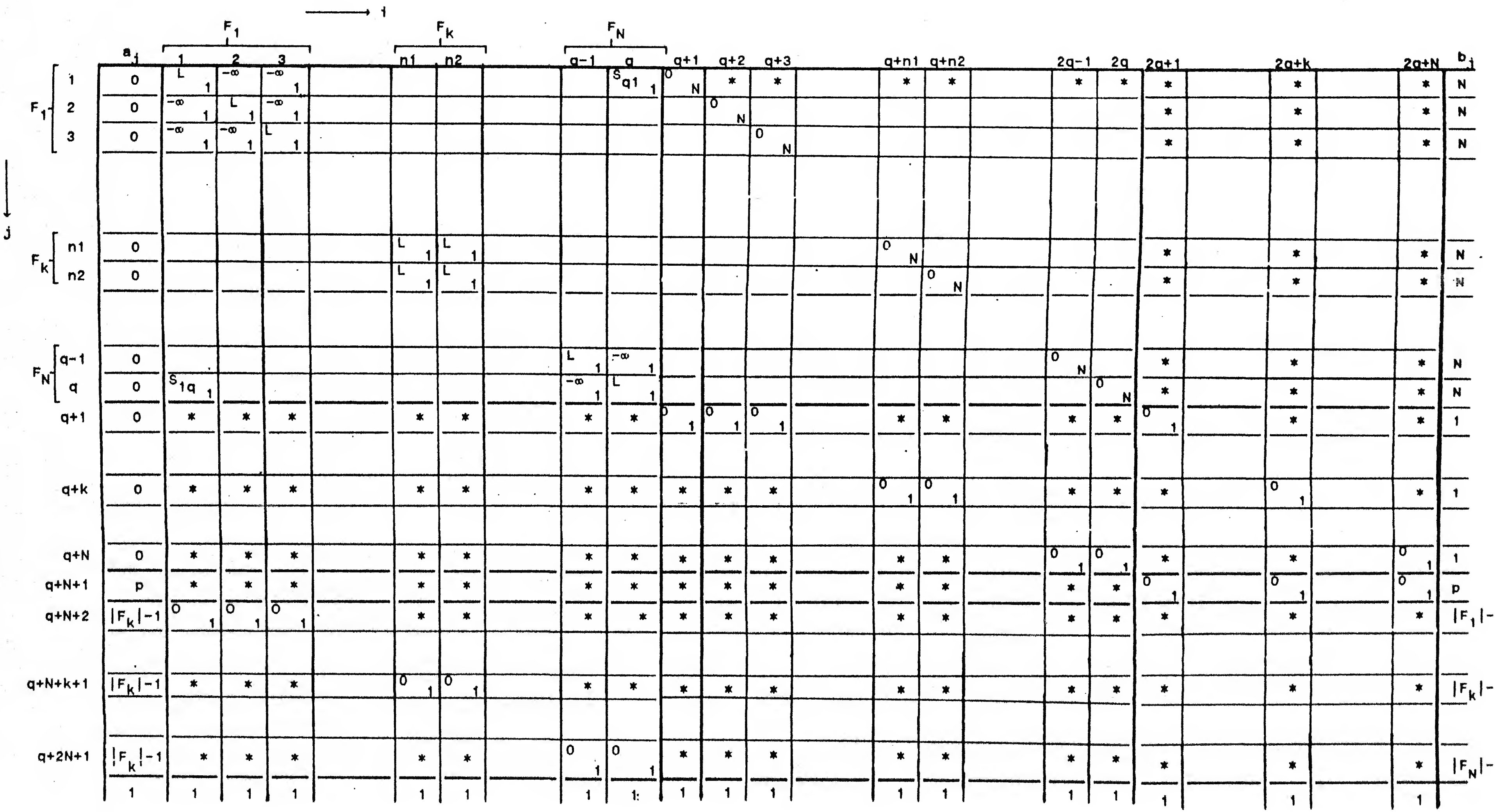


Figure 3.2: Matrix Representation of the Generalized Grouping Problem Formulated as GAP by Shtub (1989) (* in a cell corresponds to $R_{ij} = -\infty$ and $\gamma_{ij} = \infty$).

$v_n = 1 - u_n$, i.e.

$$v_n = \begin{cases} 1 & \text{if some process plan of part } n \text{ represents a process family} \\ 0 & \text{otherwise} \end{cases}$$

$$y_j = \begin{cases} 0 & \text{if process plan } j \text{ also represents process family} \\ 1 & \text{otherwise} \end{cases}$$

$z_j = 1 - y_j$, i.e.

$$z_j = \begin{cases} 1 & \text{if process plan } j \text{ also represents process family} \\ 0 & \text{otherwise} \end{cases}$$

$$w_i = \begin{cases} 1 & \text{if process plan } i \text{ is not assigned to any process family} \\ 0 & \text{otherwise} \end{cases}$$

In the definitions given above, u_n is complimentary to v_n and y_j to z_j. Further, it can be seen that the above variables and x_{ij} are related in the following manner.

$$v_n = 1 \Leftrightarrow x_{jj} = 1 \text{ for some } j \in F_n$$

and $$v_n = 0 \Leftrightarrow x_{jj} = 0 \text{ for all } j \in F_n.$$

$$y_j \equiv x_{jj}.$$

$$w_i = 1 \Leftrightarrow \sum_{j=1}^{q} x_{ij} = 0$$

and $$w_i = 0 \Leftrightarrow \sum_{j=1}^{q} x_{ij} = 1.$$

The model obtained after translating the objective function and the constraints of Figure 3.2 is denoted as M2 for further discussions and is presented below.

MODEL M3.2

OBJECTIVE FUNCTION:

$$\text{Minimize } \sum_{i=1}^{q} \sum_{j=1}^{q} \hat{s}_{ij} x_{ij}$$

The similarity coefficients $\hat{s}_{ij}$ in the objective function expression are similar to s_{ij} of the model M3.1 and also shown in the matrix of the Figure 3.2. The two similarity coefficients $\hat{s}_{ij}$ and s_{ij} are related as,

$$\hat{s}_{ij} = -s_{ij} \qquad i \neq j;\ i,j = 1, \ldots, q$$

$$\hat{s}_{jj} = -L \qquad j = 1, \ldots, q$$

where $L > \max_{i,j} (s_{ij})$.

The formulation to be presented considers the GAP as a maximization problem, hence the negative of $\hat{s}_{ij}$, i.e. s_{ij} will be used in the objective function. The objective function is thus written as:

$$\text{Maximize } \sum_{i=1}^{q} \sum_{\substack{j=1 \\ j \neq i}}^{q} s_{ij} x_{ij} + L \sum_{j=1}^{q} x_{jj} \qquad (3.9)$$

The objective function in the matrix (Figure 3.2) is read over the entries corresponding to the rows 1 to q and the columns

1 to q as the entries of similarity coefficients elsewhere have the values equal to zero.

CONSTRAINTS:

(i) <u>Selection of only one process plan for a part</u>

The constraint of one and only one process plan of a part to be included in the group formation is viewed in the matrix structure of Figure 3.2 as an equivalent constraint that the number of process plans of this part not to be included in the process families should be equal to $|F_n| - 1$. This constraint corresponds to rows $(q + N + 2)$ to $(q + 2N + 1)$ and can be represented as:

$$F_n - 1 \leq \sum_{i \in F_n} w_i \leq F_n - 1 \qquad n = 1, \ldots, N. \qquad (3.10)$$

In Figure 3.2, the columns 1 to q represent the constraint that a process plan, if selected, will be assigned to only one process family. This constraint can be written as:

$$\sum_{j=1}^{q} x_{ij} + w_i = 1 \qquad \forall\ i \in F_n\ ;$$

$$n = 1, \ldots, N. \qquad (3.11)$$

The constraints (3.10) and (3.11) together ensure that only one process plan for each part is selected in the group formation.

(ii) <u>Formation of required number of groups</u>

The constraint that exactly p process families be formed is represented by the row $(q + N + 1)$, and is:

$$p \leq \sum_{n=1}^{N} v_n \leq p. \qquad (3.12)$$

(iii) Assignment of process plans to groups

To a process family, the number of plans which can get assigned varies from 0 to N. This is represented by the rows 1 to q of the matrix and is written as:

$$0 \leq \sum_{i=1}^{q} x_{ij} + Ny_j \leq N \qquad j = 1, \ldots, q. \qquad (3.13)$$

In case when the number of process plans assigned to a process family j is zero, then plan j does not represent a process family. On the other hand, if the number of process plans assigned to a process family j is at least one, then plan j represents process family j. This aspect is represented by the columns (q + 1) to (2q) and can be stated as:

$$y_j + z_j = 1 \qquad j = 1,\ldots, q. \qquad (3.14)$$

The constraint that a process plan j chosen to represent a process family belongs to a part n, and only one plan of part n at most can represent a process family, is represented by the rows (q + 1) to (q + N) and is written as:

$$0 \leq \sum_{j \in F_n} z_j + u_n \leq 1 \qquad n = 1,\ldots, N. \qquad (3.15)$$

Further, a part through some of its process plan, may or may not represent a process family. This is given by columns (2q + 1) to (2q + N) and is written as:

$$u_n + v_n = 1 \qquad n = 1,\ldots, N. \qquad (3.16)$$

Table 3.1: Model M3.2 and Shtub's Matrix Representation of Generalized Grouping Problem.

Model M3.2	Shtub's Matrix (Figure 3.2)			
	Rows		Columns	
	From	To	From	To
<u>Objective Function</u>				
(3.9)	1	q	1	q
<u>Constraints</u>				
(a) Selection of only one process plan for a part				
(3.10)	q+N+2	q+2N+1	-	-
(3.11)	-	-	1	q
(b) Formation of required number of groups				
(3.12)	q+N+1	q+N+1	-	-
(c) Assignment of process plans to groups				
(3.13)	1	q	-	-
(3.14)	-	-	q+1	2q
(3.15)	q+1	q+N	-	-
(3.16)	-	-	2q+1	2q+N

(iv) Decision variables

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, q$$

$$u_n, v_n = 0 \text{ or } 1 \qquad n = 1, \ldots, N$$

$$y_j, z_j, w_j = 0 \text{ or } 1 \qquad j = 1, \ldots, q. \qquad (3.17)$$

Table 3.1 summarizes as how the objective function and the constraints given above are to be read over the matrix shown in Figure 3.2.

3.3 OBSERVATIONS ON THE FORMULATIONS OF KUSIAK AND SHTUB

In this section, certain observations are made on the models M3.1 and M3.2 regarding the approach for the formation of groups of parts and machines, the problem formulation and the inputs required. The observations are analyzed and limitations are also discussed. The consequent modifications and improvements are presented in the subsequent sections.

For a better understanding of the related issues, the observations are presented in the order of: grouping approach, formulation and the input requirements.

3.3.1 Grouping Approach

The solutions to the grouping provided by the models M3.1 and M3.2 are the groups of the process plans where each plan included in the group corresponds to a different part. The machine cells are to be determined thereafter. Thus, the models resume hierarchical approach for the grouping of parts and machines.

The machine cells can be determined by first assigning the machines to the process families that require those machines exclusively. A machine required by more than one process family

can be assigned one each to these families if sufficient copies are available. In case of limited number of copies, the machines can be assigned to the families that require the machine most in terms of the number of process plans using it. In this case, the groups will not be disjoint and the intercell movements will occur. The movements, of course, can be avoided by providing additional copies of such machines to the families requiring it.

3.3.2 Formulation

The discussions on the formulations of the models M3.1 and M3.2 are made with respect to the similarity coefficients used in the objective function and the constraint related to the process plan assignment. The importance of the machine capacity is also discussed from the view point of generalized grouping although it is not originally incorporated in the models of Kusiak and Shtub.

3.3.2.1 GROUP DISJOINTEDNESS

The nature of the objective function and the constraints used in the two models do not guide the selection of the process plans with the view of forming groups having minimum exceptional elements.

To illustrate the above, let us consider an example (Example 3.1) with the data as given in Table 3.2.

The optimal solution has the objective function value equal to 10, and the following process families:

Process family 1 = {1, 4}

and Process family 2 = {6, 9}.

The corresponding part families, therefore, will be:

Part family 1 = {1, 2}

and Part family 2 = {3, 4}.

Table 3.2: Data For Example 3.1.

M = 5; N = 4; p = 2; q = 9

Part (n)	Alternative Process Plan ($i \in F_n$)	Machine Type 1	2	3	4	5
1	1	1	1	1	1	
	2	1	1	1		
	3		1	1		
2	4	1	1	1	1	
	5	1		1		
3	6			1	1	1
	7				1	1
4	8					1
	9			1	1	1

Further, the corresponding machine cells on assigning all the machines required by the process (or part) families, will be:

Machine cell 1 = {1, 2, 3, 4}

and Machine cell 2 = {3, 4, 5}.

It is obvious that the two machine cells given above will require two copies of the machine types 3 and 4 for the perfect grouping. In case when only one copy of each of these machines is available, then by assigning the machines according to the requirement, the machine cells will be: {1, 2, 3, 4} and {5}, or {1, 2} and {3, 4, 5}. In these two cases, the machine cells are not disjoint and intercell movement will occur.

An interesting observation about the above example can be made regarding the objective function value and the group formation. Keeping the part families as the same as given by the

models, but choosing somewhat different process plans to form some different process families such as {2, 5} and {7, 8}, the resulting machine cells will be {1, 2, 3} and {4, 5}. This solution provides disjoint groups and requires only one copy of each machine, but has the value of the objective function as 8. This leads to the conclusion that the objective functions in the models M3.1 and M3.2 do not adequately take into consideration the issues of assignment of the machines and disjointedness of the groups, and thus the relevance of such objective function for grouping problem needs to be investigated.

3.3.2.2 COMPUTATION OF SIMILARITY

Similarity of a process plan with itself

The similarity coefficients of process plans with themselves appear as diagonal entries s_{ii} in similarity coefficient matrix $[s_{ij}]$, and as per the definition given in the equation (3.3) are zero. Therefore, the assignments made along the diagonal (x_{jj} = 1) will not contribute anything towards the objective function value of the model M3.1.

Moreover, from the equation (3.6), that requires exactly p assignments to be made along the diagonal, it can be seen that as long as the diagonal entries have the same value (even other than zero), the grouping solution will remain the same.

However, for the model M3.2, the similarity coefficients L in the diagonal must be greater than $\max_{i,j} (s_{ij})$ to ensure that p assignments are made along the diagonal of the similarity coefficient matrix.

<u>Similarity between two different process plans of the same part</u>

For the pairs of two different process plans of the same part, the values of the similarity coefficient, as given by the equation (3.2), are $-\infty$. The constraint (3.5) of the model M3.1 and the constraints (3.10) and (3.11) of the model M3.2 ensure that only one process plan from the several alternatives is selected. Therefore, the similarity coefficient for the pairs of different process plans of the same part can be allowed to assume different values without affecting the optimal solution. Hence, for such process plans the similarity coefficients can be determined using the equation (3.1) meant for the pair of process plans of two different parts. However, assignment of $-\infty$ value to the similarity between such pairs of the process plans will lead to a faster convergence towards optimal solution.

3.3.2.3 PROCESS PLAN ASSIGNMENT CONSTRAINT

As mentioned earlier, the number of assignments along the diagonal of the similarity coefficient matrix $[s_{ij}]$ is equal to p. Thus, the remaining (N-p) assignments have to be made corresponding to some non-diagonal elements of the matrix in a manner such that these (N-p) assignments appear in only those p columns (in model M3.1) or p rows (in model M3.2) that represent the process families and have assignments at their cells corresponding to the diagonal of the matrix. These observations can be used to simplify the process plan assignment constraint (3.7) of the model M3.1, and the constraints (3.12), (3.15) and (3.16) of M2. The reformulations of the models M3.1 and M3.2 after simplifying the respective constraints are presented in Section 3.4.

3.3.2.4 CONSIDERATION OF MACHINE CAPACITY

The capacities of the machines and the processing times of the operations of the various parts are not considered in the models M3.1 and M3.2 while deciding the groups of process plans.

In the generalized case, the optimal grouping selects for each part a single process plan from their alternatives. The plans thus included in the groups provide the details of the machines to be used for processing operations of the various parts. The total processing requirement on some machines which are to be used for performing the operations of these selected process plans, may become more than their available capacities. To overcome this, the easiest but expensive option of increasing the number of such machines can be adopted. However, the addition of the extra copies of the achines may hopefully be avoided by selecting certain other process plans for some parts so that the total load on all the machines is within their capacities. This may change the group formation, and also the objective function value given by the expression (3.4).

In view of the above, it seems desirable to include machine capacity as one of the constraints in the generalized grouping problem. This, of course, would require additional inputs regarding the processing times of the operations and the capacity of the machines. A formulation considering the machine capacity is attempted in Section 3.5.

3.3.3 Input Requirement

The matrix $[a_{im}]$ is the key input to the models M3.1 and M3.2. It needs a closer look as how the information related to the operations and the number of identical machines are incorpor-

ated into the models M3.1 and M3.2. The discussions related to the input requirement are made in the following subsections.

3.3.3.1 OPERATION AGGREGATION

The process plan machine matrix $[a_{im}]$, used as input in the model, simply accounts for the use of a machine by a process plan irrespective of the number of operations requiring it and the sequence in which the machine is used. Thus, each of the process plans shows simply an aggregated requirement of the corresponding part on the various types of machines.

In many practical situations, it can be commonly seen that the various operations of a part although requiring the same machine may differ from each other because of the requirements of different tools, jigs, fixtures, pallets or cutting parameters. A process plan containing these kind of details usually considers such operations to be different. Under these situations, the measure of commonality s_{ij} used in the models M3.1 and M3.2, and computed based on the aggregated input information, will not suffice. However, the problem of finding process families can be handled by the same models using in the objective function the relative requirement compatibility (RRC) as a measure of commonality. RRC, in its definition (given in detail in the Chapter II) uses explicitly the information about the number of operations in a process plan requiring common machines.

3.3.3.2 AGGREGATION OF MULTIPLE COPIES OF A MACHINE TYPE

In the incidence matrix $[a_{im}]$, the machines apparently are assumed to be of different types. The multiple copies of a machine type, if exist, can be interpreted to have been accounted for by representing them by a single machine of the same type.

Further, it is not uncommon to see on shop floors various technologically identical machines having the same or sometimes even the different capacity. Modeling the grouping problem by aggregating the multiple copies of the identical machines is equivalent to developing a model considering a single composite machine of the same type with capacity equal to the sum of the capacity of these individual machines. The models M3.1 and M3.2 can be interpreted to have accounted for the multiple copies of the same type in the above manner. The consequences of such aggregation of the machines on the grouping solution, given by these models, will be discussed latter in Section 3.5.

3.4 IMPROVEMENTS IN THE FORMULATIONS OF KUSIAK AND SHTUB

Based on the observations made in the Section 3.3, this section presents improvements over the formulations of Kusiak and Shtub. The improvements are mostly in terms of the reduced number of constraints and decision variables.

3.4.1 Improvements in Kusiak's Formulation

In view of the observations made in the Section 3.3.2.3, a simplification of the process plan assignment constraint is presented.

As observed that besides having p assignments along the diagonal of similarity coefficient matrix, the remaining (N-p) assignments are to be made in nondiagonal cells corresponding to those columns of the matrix which have assignments in the diagonal cells. These observations can be described by the following constraints.

The total nondiagonal assignments being equal to (N-p) is

stated as:

(a) $$\sum_{i=1}^{q} \sum_{\substack{j=1 \\ j \neq i}}^{q} x_{ij} = n - p. \tag{3.18}$$

At least one assignment to a process plan j ensures that the index j does represent a process family, and vice versa. These are stated as:

(b) $$\sum_{i=1}^{q} x_{ij} \geq 1 \Rightarrow x_{jj} = 1 \qquad j = 1, \ldots, q$$

and

(c) $$x_{jj} = 1 \Rightarrow \sum_{i=1}^{q} x_{ij} \geq 1 \qquad j = 1, \ldots, q.$$

The last two constraints are equivalent to the following two dichotomies.

(b') $$\sum_{i=1}^{q} x_{ij} < 1 \quad \text{or} \quad x_{jj} = 1 \quad \text{or both} \qquad j = 1, \ldots, q$$

and

(c') $$x_{jj} \neq 1 \quad \text{or} \quad \sum_{i=1}^{q} x_{ij} \geq 1 \quad \text{or both} \qquad j = 1, \ldots, q.$$

The above dichotomies can also be written as:

(b") $$\sum_{i=1}^{q} x_{ij} \leq 0 \quad \text{or} \quad x_{jj} = 1 \quad \text{or both} \qquad j = 1, \ldots, q \tag{3.19}$$

and

(c") $$x_{jj} \leq 0 \quad \text{or} \quad \sum_{i=1}^{q} x_{ij} \geq 1 \quad \text{or both} \qquad j = 1, \ldots, q. \tag{3.20}$$

The above dichotomies can be further rewritten in the usual form of the constraints according to the scheme given by Garfinkel and Nemhauser (1972). It has been shown by them that a dichotomy, $f(x) \le 0$ or $g(x) \ge 0$ or both, can be written as:

$$f(x) \le \theta \bar{f}$$

$$g(x) \ge (1 - \theta)\underline{g}$$

where θ is a 0-1 variable, $\bar{f}$ is an upper bound on the value of $f(x)$, and $\underline{g}$ is a lower bound on the value of $g(x)$.

As mentioned earlier, out of the q process families, only the p process families will have some process plans assigned to them, and remaining (q - p) families will have no assignments. Thus, to a process family either no assignment takes place or all $\left\{N-(p-1)\right\}$ process plans, each corresponding to different parts, get assigned. In case of no assignment to process plan j, $\sum_{i=1}^{q} x_{ij}$ will be equal to zero. And, for the case of the maximum assignment to process plan j, $\sum_{i=1}^{q} x_{ij}$ will be equal to $\left\{N-(p-1)\right\}$. Therefore, the lower and upper bounds on the values of $\sum_{i=1}^{q} x_{ij}$ will be zero and (N-p+1), respectively. The respective lower and upper bounds for x_{ij}, as predefined, are zero and one. Based on this information on the bounds and the method of breaking dichotomy into usual form of constraints as described earlier, the dichotomy (3.19) can be written as:

$$\sum_{i=1}^{q} x_{ij} \le (N - p + 1)\rho_j \qquad j = 1, \ldots, q \qquad (3.21)$$

$$x_{jj} - 1 \ge (-1)(1 - \rho_j) \qquad j = 1, \ldots, q \qquad (3.22)$$

where $\rho_j \in \{0, 1\}$ $\qquad j = 1, \ldots, q.$

The above two constraints are equivalent to the following.

$$\sum_{i=1}^{q} x_{ij} \le (N - p + 1)\rho_j \le (N - p + 1)x_{jj}$$

$$j = 1, \ldots, q$$

The above inequalities can be reduced to

$$\sum_{i=1}^{q} x_{ij} \le (N - p + 1)x_{jj} \qquad j = 1, \ldots, q \qquad (3.23)$$

It can be seen that the number of the constraints in the expression (3.23) is less as compared to that given in the expressions (3.21) and (3.22) which are obtained after breaking the dichotomy (3.19). Further, the constraint (3.23) is free of the variable ρ_j.

Similarly, the dichotomy (3.20) can finally be simplified to:

$$\sum_{i=1}^{q} x_{ij} \ge x_{jj} \qquad j = 1, \ldots, q.$$

The above constraint is equivalent to the following.

$$\sum_{\substack{i=1 \\ i \ne j}}^{q} x_{ij} \ge 0 \qquad j = 1, \ldots, q$$

The above constraint is a trivial constraint and thus need not be considered. Further, the constraint (3.18) in view of the constraints (3.5) and (3.6) is also not considered. Therefore, the constraint (3.7), related to the assignment of process plans to the process families, can be substituted by the constraint (3.23).

The improved formulation (to be referred to as model M3.3) is given below.

MODEL M3.3

$$\text{Maximize} \quad \sum_{i=1}^{q} \sum_{j=1}^{q} s_{ij} x_{ij}$$

Subject to (3.5), (3.6), (3.23) and (3.8).

Table 3.3 presents the details related to the number of variables and constraints used by the models M3.1, M3.2 and M3.3, and also by the model M3.4 that will be described later. From the table, it can be seen that for all practical grouping problems, the number of constraints in the model M3.2 will be less as compared to that of M3.1. However, the number of the decision variables will always be more. The model M3.3, requires the least

Table 3.3: Number of Constraints and Variables for Different Models.

	Model			
	M3.1	M3.2	M3.3	M3.4
1. Number of constraints:				
(a) with alternative process plans	q^2+N+1	4q+4N+1	q+N+1	3q+N+1
(b) without alternative process plans	N^2+N+1	4N+1	2N+1	3N+1
2. Number of variables:				
(a) with alternative process plans	q^2	q^2+3q+2N	q^2	q^2+2q
(b) without alternative process plan	N^2	N^2+2N	N^2	N^2+N
3. Size of matrix:				
(a) with alternative process plans	-	(q+2N+1)(2q+N)	-	(q+N)(q+1)
(b) without alternative process plans	-	(N+1)(2N)	-	(N)(N+1)

number of constraints compared to both the models M3.1 and M3.2, and involves the same number of variables as for M3.1. Thus, the model M3.3 would contain for sure, the least number of constraints and decision variables.

3.4.2 Improvements in Shtub's Formulation

The formulation given in Section 3.2.2 can be simplified by combining together the process plan assignment constraints (3.15) and (3.16) with the group formation constraint (3.12). The resulting constraint after simplification will be:

$$-N + p \leq \sum_{n=1}^{N} \sum_{j \in F_n} z_j \leq p.$$

The inequality on the left side of the above constraint is trivial because (-N+p) is a negative number and $\sum_{n=1}^{N} \sum_{j \in F_n} z_j$ can at least be equal to zero. Further, to ensure that out of q process plans p plans represent the required number of p process families, the lower bound on the value of the expression $\sum_{n=1}^{N} \sum_{j \in F_n} z_j$ in the above constraint must be equal to p. The constraint, therefore, can be written as:

$$p \leq \sum_{n=1}^{N} \sum_{j \in F_n} z_j \leq p$$

The above constraint is equivalent to the group formation constraint (3.6) of models M3.1 and M3.3 with z_j being equivalent to x_{jj}. This constraint and the discussions made in the above paragraph suggests that the formulation in the model M3.2 can be

made more tighter by having lower bound value of the expression $(\sum_{j\in F_n} z_j + u_k)$ in the constraint (3.15) also equal to one instead of zero which is trivial and comes directly from the definition of z_j and u_k. The above constraint can be combined with the constraint (3.14) to yield the following:

$$q - p \le \sum_{n=1}^{N} \sum_{j\in F_n} y_j \le q - p. \tag{3.24}$$

The simplified formulation for the model M3.2, to be called as model M3.4 for further references, is given below.

<u>MODEL M3.4</u>

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} s_{ij} x_{ij}$$

Subject to: (3.10), (3.11), (3.13), (3.24), (3.8) and (3.17).

A comparison of the total number of variables and constraints required by the models M3.2 and M3.4 are also presented in the Table 3.3. Figure 3.3 shows the assignment matrix corresponding to the model M3.4. It can be seen from the Figures 3.2 and 3.3 that the size of the matrix for the model M3.4 is smaller as compared to that for the model M3.2.

Table 3.4 summarizes as how the objective function and the constraints of the model M3.4 are to be read over the matrix shown in the Figure 3.3.

It should be noted that the entries at the right bottom corner of each cell in the matrix shown in Figures 3.2 and 3.3 are to be used for reading constraints along rows. However, for all

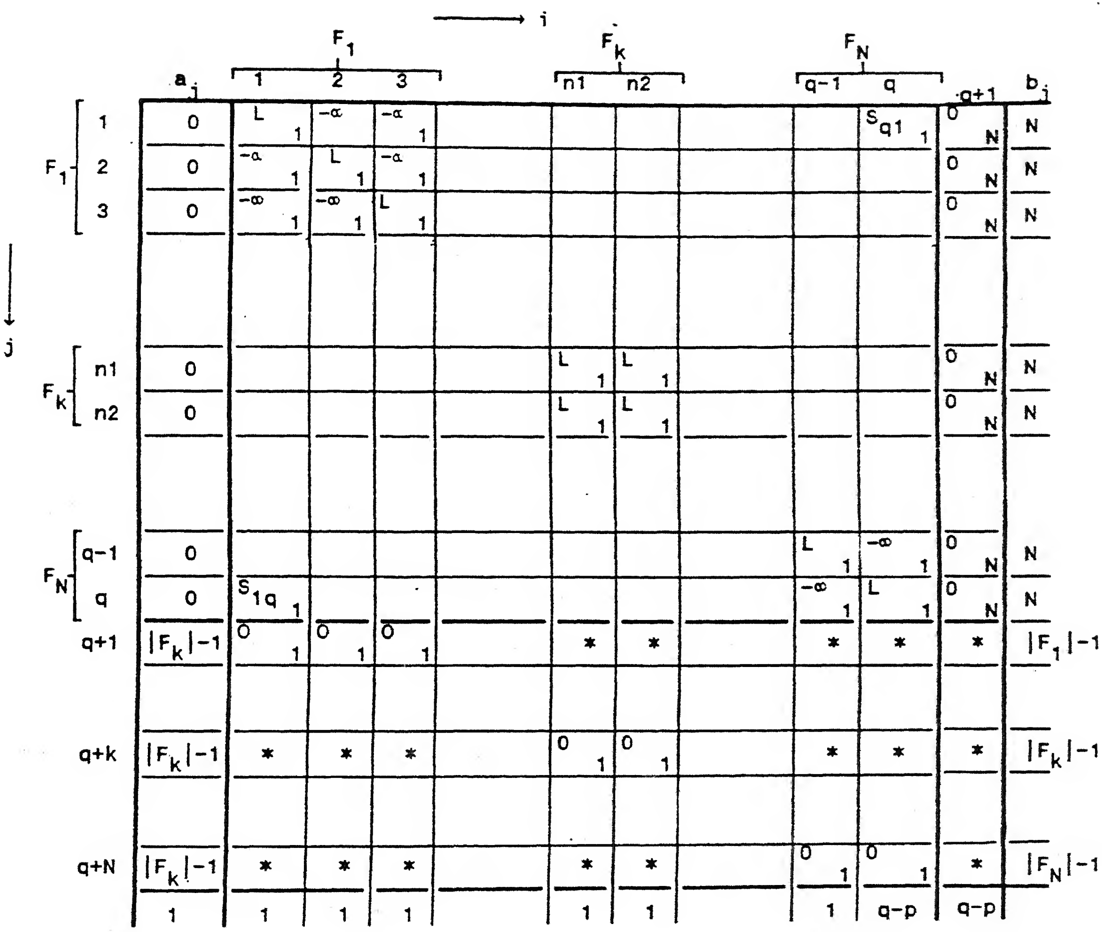


Figure 3.3: Matrix Representation of the Grouping Problem Corresponding to Model M3.4.

the constraint along columns, these number are to be taken as one.

Further, it can be seen that the constraints of the model M3.4 and their combinations are equivalent to the constraints of the model M3.3. The process plan selection constraints (3.10) and (3.11) of the model M3.4 can be combined together to yield the constraint (3.5)

Table 3.4: Correspondence Between Model M3.4 and Matrix Representation Given in Figure 3.3.

Model M3.4	Matrix Representation (Figure 3.3)			
	Rows		Columns	
	From	To	From	To
<u>Objective Function</u>				
(3.9)	1	q	1	q
<u>Constraints</u>				
(a) Selection of only one process plan for a part				
(3.10)	q+1	q+N	-	-
(3.11)	-	-	1	q
(b) Formation of required number of groups				
(3.24)	-	-	q+1	q+1
(c) Assignment of process plans to groups				
(3.13)	1	q	-	-

of the model M3.3. The constraint (3.24) is the complement of the constraint (3.6) in the model M3.3. The constraint (3.24) ensures that out of q process plans (q-p) ones do not represent the process families, whereas the constraint (3.6) states that exactly

p process plans represent process families. Therefore, the constraint (3.24) and (3.6) can be considered to be equivalent. Based on this, it can be concluded that complement of y_j, i.e. $(1-y_j)$ is equivalent to x_{jj}. The process plan assignment constraint (3.13) of the model M3.4 can be reduced to the constraint (3.23) of the model M3.3 as can be seen from the subsequent deliberations.

The lower bound on the values of the expression $\left\{\sum_{i=1}^{q} x_{ij}+Ny_j\right\}$ in the constraint (3.13) is trivial because the lowest value of this expression from the definition of x_{ij} and y_j is any way equal to zero. Therefore, the constraint (3.13) is equivalent to:

$$\sum_{i=1}^{q} x_{ij} + Ny_j \leq N \qquad j = 1, \ldots, q$$

or

$$\sum_{i=1}^{q} x_{ij} \leq N(1-y_j) \qquad j = 1, \ldots, q \tag{3.25}$$

Since $(1-y_j)$ is shown to be equivalent to x_{jj}, the constraint (3.25) can be written as:

$$\sum_{i=1}^{q} x_{ij} \leq N\, x_{jj} \qquad j = 1, \ldots, q. \tag{3.26}$$

The constraint (3.26) suggests that the upper bound on the number of process plans that can be assigned to a process family j is N. However, as discussed earlier in the Section 3.4.1, the number of process plans which can be assigned to a family is, in fact, less than N and is at most equal to (N - p + 1). Thus, the constraint (3.23) of the model M3.3 offers tighter bound as compared to that by the constraint (3.26).

3.5 EXTENSIONS

This section presents extensions of the basic generalized grouping problem considered in the models M3.1 and M3.2. The extensions encompass certain generalities of the problem environment as observed and analyzed in the Section 3.3. For the generalizations, the structure of the model M3.3 is used which, as discussed in Section 3.4, involves lesser number of variables as compared to the models M3.2 and M3.4, and has lesser number of constraints as compared to the models M3.1, M3.2 and M3.4.

For the generalizations considered, the basic constraints, such as selection of only one process plan out of the several alternative plans for each of the parts given by (3.5), formation of exactly p number of groups given by (3.6), and assignment of process plans given by (3.23), are retained. In addition, some more constraints are appended to suit the grouping requirements.

As pointed out earlier in the Section 3.3.3.1, the similarity measure s_{ij} used in the objective function expression (3.4) does not distinguish the operations that require the same type of machine. However, for the purpose of grouping which also aims at minimizing the movements of the parts, it is necessary to consider the operations requiring the same type of machines distinctly and separately because all such operations may not be carried out on a machine type in one pass itself. This clearly prohibits the use of the similarity measure s_{ij} in generalized grouping. As mentioned in the Section 3.3.3.1, the relative requirement compatibility (RRC) that considers the association between process plans recognizing such distinction about the operations, can be used for this purpose. The association between a pair of process plans, in

general, can be measured in terms of the number of common machines, the number of operations that can be performed on the common machines or the operation processing times on the common machines. Using these, various expressions for the relative requirement compatibility R_{ij}, measuring the compatibility of a process plan i with respect to some plan j, are written as:

$$\text{(i)}\ R_{ij} = \frac{\text{Number of machine types common to process plans i and j}}{\text{Number of machine types required by process plan i}},$$

$$\text{(ii)}\ R_{ij} = \frac{\begin{array}{c}\text{Number of operations in process plan i requiring}\\ \text{machines of the type common to plan j}\end{array}}{\text{Total number of operations in process plan i}}$$

and

$$\text{(iii)}\ R_{ij} = \frac{\begin{array}{c}\text{Total processing time of operations in process plan i}\\ \text{requiring machines of the type common to plan j}\end{array}}{\text{Total processing time of operations in process plan i}}$$

The above definitions of R_{ij} can also be used for determining association between two route plans i and j.

A process plan, in general, specifies the type of machine against each of the operations which can be used for processing them. A route plan, on the other hand, contains the specific detail as to which particular machine a part will go for an operation.

For the generalized grouping, among three definitions of RRC given above the last two will be more suitable because these identify each operation separately. However, depending upon the grouping requirements and the nature of the manufacturing system, one can consider the other definitions of RRC (for detail see the

Chapter II) which are based on some combinations of required resources such as machines, pallets, jigs, fixtures, cutting tools, etc.. For further deliberations, the RRC based on the number of operations or the processing times is taken as the representative compatibility measure between a pair of process/route plans.

The second definition of RRC will be suitable where the desirability is to have groups in which the majority of the operations of the member process plans require the machines that are common to a process plan that represents of the corresponding process family. Similarly, the third definition can be found suitable for the case where one is interested in determining the groups when most of the processing load of the member process plans are on the machines that are common to a plan representing the corresponding process family.

It may be noted that the use of the last two definitions of R_{ij} in the objective function will require additional input related to the detailed specification of the operations showing the required machines and corresponding processing times.

For the development of the various expressions in this section, the following notations and definitions in addition to the ones introduced in the Section 3.2, are used.

Indices

$\underline{i}$, $\underline{j}$: route plans

k, $\underline{k}$: operation

m : machine type

n, $\underline{n}$: parts

Parameters

$\underline{g}$ = the total number of route plans

M = the total number of machines

f_m = cost of single machine of machine type m

G_n = set of route plans for part n

K_n = the total number of operations to be performed on part n

M_L = the minimum number of machines that must be assigned to a machine cell

M_T = the total number of machines that can be assigned over all the machine cells

M_U = the maximum number of machines that can be assigned to a machine cell

N_m = set of machines of type m

P_L = the minimum number of parts that must be assigned to a part family

P_U = the maximum number of parts that can be assigned to a part family

$T_{\underline{m}}$ = capacity of machine $\underline{m}$

t_{ik} = processing time for k^{th} operation in process plan i

$t_{\underline{i}k}$ = processing time for k^{th} operation in route plan $\underline{i}$

B(i, k) = type of machine required for processing k^{th} operation in process plan i

$\underline{B}(\underline{i}, k)$ = machine required for processing k^{th} operation in route plan $\underline{i}$

Metrics

$$\mu(c) = \begin{cases} 1 & \text{if } c \geq 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\lambda(m_1, m_2) = \begin{cases} 1 & \text{if machines } m_1 \text{ and } m_2 \text{ are of the same type} \\ 0 & \text{otherwise} \end{cases}$$

Derived Parameters

$$a_{im} = \begin{cases} 1 & \text{if process plan i requires machine type m} \\ 0 & \text{otherwise} \end{cases}$$

$$= \mu \left[\sum_{k=1}^{K_n} \delta \left\{ B(i, k), m \right\} \right]$$

$$a_{\underline{i},\underline{m}} = \begin{cases} 1 & \text{if route plan } \underline{i} \text{ requires machine } \underline{m} \\ 0 & \text{otherwise} \end{cases}$$

$$= \mu \left[\sum_{k=1}^{K_n} \delta \left\{ \underline{B}(\underline{i}, k), \underline{m} \right\} \right]$$

C_m = total capacity of all the machines of type m

$$= \sum_{\underline{m} \in N_m} T_{\underline{m}}$$

R_{ij} = relative requirement compatibility of process plan i with respect to process plan j

$R_{\underline{i},\underline{j}}$ = relative requirement compatibility of route plan $\underline{i}$ with respect to route plan $\underline{j}$

t_{im} = total processing time requirement of process plan i on machine type m

$$= \sum_{k=1}^{K_n} t_{ik} \, \delta \left\{ B(i, k), m \right\}$$

$t_{\underline{i},\underline{m}}$ = total processing time requirement of route plan $\underline{i}$ on machine $\underline{m}$

$$= \sum_{k=1}^{K_n} t_{\underline{i},k} \, \delta \left\{ \underline{B}(\underline{i}, k), \underline{m} \right\}$$

Decision Variables

$$x_{\underline{i},\underline{j}} = \begin{cases} 1 & \text{if route plan } \underline{i} \text{ is assigned to route family } \underline{j} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{mj} = \begin{cases} 1 & \text{if all the machines of type m are assigned to process family j} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{\underline{m},\underline{j}} = \begin{cases} 1 & \text{if machine } \underline{m} \text{ is assigned to route family } \underline{j} \\ 0 & \text{otherwise} \end{cases}$$

z_{mj} = number of machines of type m assigned to process family j

As mentioned earlier, in addition to the basic constraints of the model M3.3, some more constraints such as the constraints on the machine capacity, disjointedness of the groups, etc. are appended depending upon the grouping requirements. The expressions for the constraints will depend upon as how the information related to the number of identical machines and their capacities are incorporated into the formulations. There are two possible ways in which such information can be provided:

(i) Identical machines are not considered separately and are rather indexed by the corresponding machine type, and

(ii) Identical machines are considered separately and are indexed explicitly.

The formulations of the grouping problems for the above mentioned two scenarios are given in the following subsections.

3.5.1 Machines Denoted by Their Types

For this case, the machines are categorized based on their types, or in other words, based on their capabilities. The multiple copies of a machine type can be treated in the following two ways for the purpose of grouping.

(i) Multiple copies of a machine type are assigned to only one group.

(ii) The multiple copies of a machine type are considered for assigning to more than one group.

The formulation for these two different situations are presented below.

3.5.1.1 IDENTICAL MACHINES ASSIGNED TO A SINGLE GROUP

In this case, the identical machines are considered combined together by representing them all by a single composite machine of the same type with capacity equal to the sum of their individual capacities. Consequently, all the identical machines will go to the group to which the corresponding composite machine get assigned.

The expressions for the objective function and the various constraints are as follows.

OBJECTIVE FUNCTION

The objective of maximizing the total relative requirement compatibility can be written as:

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij}\, x_{ij} \qquad (3.27)$$

where R_{ij} $(i \in F_n)$ is calculated using any of the following

expressions.

$$\text{(i)}\quad R^1_{ij} = \frac{\sum_{k=1}^{K_n} a_{j,B(i,k)}}{K_n} \tag{3.28}$$

$$\text{(ii)}\quad R^2_{ij} = \frac{\sum_{k=1}^{K_n} t_{ik}\cdot a_{j,B(i,k)}}{\sum_{k=1}^{K_n} t_{ik}} \tag{3.29}$$

R^1_{ij} is computed based on the number of operations, while R^2_{ij} on the basis of processing times.

CONSTRAINTS

(i) Selection of only one process plan for a part

This constraint is the same as (3.5) in the model M3.3.

(ii) Formation of required number of groups

This can be represented by the constraint (3.6) of the model M3.3.

(iii) Assignment of process plans to groups

This constraint is the same as (3.23) in the model M3.3.

(iv) Number of parts assigned to a group

As mentioned in the Section 3.1, the generalized grouping problem should accord cognizance to the number of parts to be assigned to a group. The constraints on the maximum and the minimum number of parts in a group can be represented as:

$$\sum_{i=1}^{q} x_{ij} \leq P_U \qquad j = 1, \ldots, q \tag{3.30}$$

and $$\sum_{i=1}^{q} x_{ij} \geq P_L \, x_{jj} \qquad j = 1, \ldots, q. \quad (3.31)$$

(v) <u>Load on machine</u>

Using the definition of x_{ij} and taking into consideration the constraint (3.5), as given in the Section 3.2.1, following holds:

$$\sum_{j=1}^{q} x_{ij} = \begin{cases} 1 & \text{if process plan } i \text{ is assigned to some process family} \\ 0 & \text{otherwise} \end{cases}$$

In view of the above, the constraint that the load on a machine should not exceed the available capacity, can be written as:

$$\sum_{i=1}^{q} t_{im} \sum_{j=1}^{q} x_{ij} \leq C_m \qquad m = 1, \ldots, M. \quad (3.32)$$

(vi) <u>Assignment of all the copies of a machine type to one group</u>

$$\sum_{j=1}^{q} y_{mj} \leq 1 \qquad m = 1, \ldots, M \quad (3.33)$$

(vii) <u>Group disjointedness</u>

The conditions on obtaining disjoint machine cells can be represented by the following constraint.

$$\sum_{i=1}^{q} a_{im} x_{ij} \geq 1 \Leftrightarrow y_{mj} = 1 \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M$$

The above constraint ensures that all the machines required by the process plans of a process family are available in the corresponding machine cell, and only the required machines are assigned. These two restrictions are presented by the constraints

(3.34) and (3.35), respectively.

$$\sum_{i=1}^{q} a_{im}x_{ij} \geq 1 \Rightarrow y_{mj} = 1 \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M \qquad (3.34)$$

$$y_{mj} = 1 \Rightarrow \sum_{i=1}^{q} a_{im}x_{ij} \geq 1 \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M \qquad (3.35)$$

In the absence of the constraint (35), the solution may have a machine cell with a machine which is not required by any of the process plans of the corresponding process family. The assignment of such a machine to a cell will be possible only when the process plans of the other families do not require it. Since the grouping problem at the operational level assumes that the machines are already existing, thus the assignment of the non-required machines to some cells will have the same effect as of not assigning them to any of the cells. In both the situations, such machines will not be used and thus would remain idle.

It may be noted that such an assignment of the machines that are not required by any of the process plans included in families, though needed by some process plans not included in any of the families, is a special characteristic of the generalized grouping, and is the consequence of the selection criterion which chooses only one process plans for each part for maximizing the objection function. Such a situation, however, will never arise in the simple grouping problem.

In view of the observations made in the preceding paragraphs, the constraint (3.35) may be omitted. This may also reduce the

computational requirements.

The constraints (3.34) and (3.35) are equivalent to the following dichotomies, respectively:

(a) $\sum_{i=1}^{q} a_{im}x_{ij} < 1$ or $y_{mj} = 1$ or both $\quad j = 1, \ldots, q; \quad m = 1, \ldots, M,$

and

(b) $y_{mj} \neq 1$ or $\sum_{i=1}^{q} a_{im}x_{ij} \geq 1$ or both $\quad j = 1, \ldots, q; \quad m = 1, \ldots, M.$

These two dichotomies can be restated as:

$$(a') \quad \sum_{i=1}^{q} a_{im}x_{ij} \leq 0 \text{ or } y_{mj} = 1 \text{ or both} \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M, \tag{3.36}$$

and

$$(b') \quad y_{mj} \leq 0 \text{ or } \sum_{i=1}^{q} a_{im}x_{ij} \geq 1 \text{ or both} \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M. \tag{3.37}$$

Since it is possible to have a process family j whose any member plan may not require a machine type m, $\sum_{i=1}^{q} a_{im}x_{ij}$ will be equal to zero. Therefore, the lower bound on the value of the expression $\sum_{i=1}^{q} a_{im}x_{ij}$ will be equal to zero.

On the other hand, where each of the process families (other than family j) contains only one process plan, the remaining (N-p+1) assignments of process plans will be to process family j. Further, all the process plans assigned to family j may require machine type m. Thus, the upper bound on the value of the expression $\sum_{i=1}^{q} a_{im}x_{ij}$ will be (N-p+1).

The upper bound can, however, be more tighter for the case when there is a limit on the maximum number of parts that can be assigned to a group. For such cases, the upper bound will be equal to this maximum number. Since the respective values of lower and upper bounds on the value of y_{mj} are equal to zero and one, the dichotomy (3.36) using the method as described in the Section 3.4 (Garfinkel and Nemhauser (1972)), can be written as:

$$\sum_{i=1}^{q} a_{im}x_{ij} \leq \alpha_{mj} (N - p + 1) \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q,$$

and
$$y_{mj} - 1 \geq (1 - \alpha_{mj}) (- 1) \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q,$$

where
$$\alpha_{mj} \in \{0, 1\} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q.$$

The two constraints given above can be linked together as given below.

$$\sum_{i=1}^{q} a_{im}x_{ij} \leq (N - p + 1)\alpha_{mj} \leq (N - p + 1)y_{mj} \qquad m = 1, \ldots, M \quad j = 1, \ldots, q$$

In the above expression, there are 2Mq number of constraints and Mq number of α_{mj} variables. The expression may become free of α_{mj} and will have only Mq constraints once it is reduced to the following without loosing any meaning of the constraint.

$$\sum_{i=1}^{q} a_{im}x_{ij} \leq (N - p + 1)y_{mj} \qquad m = 1, \ldots, M \quad j = 1, \ldots, q \quad (3.38)$$

Similarly, the dichotomy (3.37) can be written as:

$$y_{mj} \leq \beta_{mj} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q,$$

and $$\sum_{i=1}^{q} a_{im}x_{ij} - 1 \geq (1 - \beta_{mj})(-1) \qquad m = 1, \ldots, M \quad j = 1, \ldots, q,$$

where $$\beta_{mj} \in \{0, 1\} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q.$$

The above two constraints, in a manner similar to that followed above, can be simplified to:

$$\sum_{i=1}^{q} a_{im}x_{ij} \geq y_{mj} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q.$$

The above constraint ensures that only required machines get assigned, whereas the constraint (3.38) will ensure that required machines are available in the corresponding machine cell.

DECISION VARIABLES

$$x_{ij} \in \{0, 1\} \qquad i = 1, \ldots, q; \quad j = 1, \ldots, q$$

$$y_{mj} \in \{0, 1\} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q \qquad (3.39)$$

MODELS

The different combinations of the three constraints, viz. the machine capacity, the group disjointedness, and the limits on the

minimum and the maximum number of parts that can be assigned to a group, will give several problem scenarios. Each of these scenarios, however, include the basic constraints, such as, the constraints on process plan selection, group formation and process plan assignment.

For each of these scenarios which has the constraints on both the machine capacity and the group disjointedness, the constraint (3.32) does not remain valid. Further, it is not advisable to consider the constraint (3.38) separately since the two constraints together can be represented by the following.

$$\sum_{i=1}^{q} t_{im} x_{ij} \leq c_m y_{mj} \qquad m = 1, \ldots, M \quad (3.40)$$

Similarly, the constraint (3.30) on the maximum number of parts that can be assigned to a group need not be considered separately in view of the process plan assignment constraint (3.23). This constraint together with the process plan assignment constraint (3.23) can be represented by the following:

$$\sum_{i=1}^{q} x_{ij} \leq \left\{ \min \left[(N-p+1),\ P_U\right] \right\} x_{jj} \qquad j = 1, \ldots, q. \quad (3.41)$$

It should be noted that the machine assignment constraint (3.33) has to be considered only when the constraint on the group disjointedness, given by (3.38), or along with the machine capacity constraint given by (3.40) is included.

For studying the influence of these constraints on grouping, the scenarios represented by the following models (M3.5(a) to M3.5(h)) are considered.

Model	Constraints aditional to (3.5),(3.6) and (3.23) on: Machine Capacity	Perfect Grouping	No. of parts in a group: Max	Min	Machine Assignment	Decision Variables
M3.5(a)	-	-	-	-	-	(3.8)
M3.5(b)	-	-	-	(3.31)	-	(3.8)
M3.5(c)	-	-	(3.41)	-	-	(3.8)
M3.5(d)	-	-	(3.41)	(3.31)	-	(3.8)
M3.5(e)	(3.32)	-	-	-	-	(3.8)
M3.5(f)	-	(3.38)	-	-	(3.33)	(3.8),(3.39)
M3.5(g)	(3.40)	(3.40)	-	-	(3.33)	(3.8),(3.39)
M3.5(h)	(3.40)	(3.40)	(3.41)	(3.31)	(3.33)	(3.8),(3.39)

As discussed previously, the constraint (3.23) is trivial in the presence of the constraint (3.41), hence need not be considered into the models M3.5(c), M3.5(d) and M3.5(h).

The numerical examples presented in Section 3.7 are used for illustrating the behaviour of the solution of these models.

3.5.1.2 IDENTICAL MACHINES DISTRIBUTED AMONG GROUPS

The aggregation of identical machines into one leads to the assignment of all the copies of a machine type to only one group. Such aggregation may lead to a high intercell movements as the parts requiring this type of machine may get assigned to different groups and, thus are to be moved for carrying out certain operations of these parts to the group possessing this machine type. It, therefore, seems advisable to distribute the multiple copies of the machines to more than one group to reduce the intercell movements.

Further, for the case of machine aggregation, in a solution obtained satisfying the requirements of group disjointedness, there will indeed be no intercell movements, but all the copies of a machine type assigned to some group may not be required by the

process plans corresponding to this group. The process of making these extra copies available for inclusion in some other groups can be viewed as equivalent to that of relaxing the consideration of machine aggregation. In general, the relaxation of constraints brings improvement in the objective function value. Therefore, the relaxation of the condition on the machine aggregation is expected to bring improvement in the objective function value while retaining the group disjointedness.

In view of the advantages discussed above, the grouping problem stated in this section considers the possibility of the assignment of a machine type to more than one group. Since each machine is not considered separately, the models discussed in this section cannot take into consideration their individual capacity. The capacity of the machines of the same type are assumed to be equal, i.e.

$$T_{\underline{m}} = T_m \qquad \forall\ \underline{m} \in N_m;\ m = 1, \ldots, M.$$

The expressions for the objective functions and the various constraints are given below.

OBJECTIVE FUNCTION

(i) Maximization of the total relative requirement compatibility

The expression for this objective remains the same as given by (3.27) in the Section 3.5.1.1.

(ii) Minimization of investment on the machines

The expression related to this objective can be written as:

$$\text{Minimize} \sum_{m=1}^{M} f_m \sum_{j=1}^{q} z_{mj}. \tag{3.42}$$

CONSTRAINTS

Various constraints along with their expressions are given below:

(i) Selection of only one process plan for a part: (3.5).

(ii) Formation of required number of groups: (3.6).

(iii) Assignment of process plans to groups: (3.23).

(iv) Number of parts assigned to a group

(a) Limit on the maximum number of parts: (3.30), or (3.41) when the constraint on the process plan assignment is combined.

(b) Limit on the minimum number of parts: (3.31).

(v) Load on machines : (3.32).

(vi) Total number of machines of each type

The total number of machines of a type which can be assigned to different machine cells should not exceed the number of available copies (the number should be referred as the permissible number of copies of a machine type in the context of production system design problem). This constraint can be written as:

$$\sum_{j=1}^{q} z_{mj} \le |N_m| \qquad m = 1, \ldots, M. \quad (3.43)$$

(vii) Group disjointedness

The constraint, that each machine cell should contain only the machines required by the corresponding process family, can be expressed as follows.

$$\sum_{i=1}^{q} a_{im} x_{ij} \ge 1 \;\Leftrightarrow\; z_{mj} \ge 1 \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M$$

The upper and lower bounds on the values of z_{mj} will be $|N_m|$

and zero, respectively. The upper and lower bound on the value of $\sum_{i=1}^{q} a_{im}x_{ij}$, as shown in the Section 3.4, are equal to (N-p+1) and zero, respectively. Therefore, following the method as described already in the Section 3.2, the above constraint can be represented by the following two constraints:

$$\sum_{i=1}^{q} a_{im}x_{ij} \leq (N - p + 1)z_{mj} \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M, \tag{3.44}$$

and

$$\sum_{i=1}^{q} a_{im}x_{ij} \geq \frac{z_{mj}}{|N_m|} \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M. \tag{3.45}$$

If the restriction that the only required machines are to be assigned to the various machine cells is relaxed, then the constraint (3.45) is not to be considered.

(viii) <u>Total number of machines in production system</u>

This constraint becomes important for the case when a new production system is to be designed. The number of machines which can be accommodated depends upon the total floor area, the layout of materials handling system, etc.. The constraint can be expressed as:

$$\sum_{m=1}^{M} \sum_{j=1}^{q} z_{mj} \leq M_T. \tag{3.46}$$

(ix) <u>Total number of machines in a group</u>

For the reasons as described in the Section 3.1, it may become necessary to consider upper and lower limits on the number of machines that can be assigned to a group. The constraints representing such limits are:

$$\sum_{m=1}^{M} z_{mj} \leq M_U \qquad j = 1,\ldots, q, \quad (3.47)$$

$$\sum_{m=1}^{M} z_{mj} \geq M_L\, x_{jj} \qquad j = 1,\ldots, q. \quad (3.48)$$

The respective constraints (3.47) and (3.48) ensure that the limits on the maximum and the minimum number of machines that can be assigned to a group are not violated.

DECISION VARIABLES

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, q$$

$$z_{mj} \geq 0 \text{ and integer} \qquad m = 1, \ldots, M; \; j = 1, \ldots, q \quad (3.49)$$

MODELS

In this case also, as discussed in the Section 3.5.1.1, the different combinations of the constraints, in addition to the basic ones, will encompass different problem scenarios.

The machine capacity constraint (3.32) simply ensures that the total load (of processing parts using their selected process plans) on a machine does not exceed the available capacity. The consideration of this constraint leads to a grouping solution which is feasible from the viewpoint of operations assignment also. However, this constraint does not ensure that the groups thus obtained will be disjoint. The group disjointedness constraint (3.44), on the other hand, ensures that to each group all those machines are assigned that are required by the process plans assigned to the same group. Such a solution may not be disjoint

from operational viewpoint because the machines assigned to a group may not be having capacities required to process all the operations of all the parts assigned to the same group, and thus may require intercell movements. Therefore, for the situations where groups are to be operationally disjoint and do not have to violate the machine capacity restrictions, the following constraint should be used instead of the constraints (3.32) and (3.44).

$$\sum_{i=1}^{q} t_{im}\, x_{ij} \le T_m\, z_{mj} \qquad j = 1, \ldots, q; \quad m = 1, \ldots, M. \tag{3.50}$$

Further, for the generalized grouping problem at operational level, it does not make sense to consider the constraint (3.45) which ensures assignment of only required machines to various machine cells. The generalized grouping problem, for this case, is as given below.

MODEL 3.6(a)

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij}\, x_{ij}$$

Subject to: (3.5), (3.6), (3.31), (3.41), (3.43), (3.46), (3.47), (3.48), (3.50), (3.8) and (3.49).

The problem of disjoint group formation, without considering the constraints on the machine capacity and on the total number of machines in the system, is represented by the following model.

MODEL 3.6(b)

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij}\, x_{ij}$$

Subject to: (3.5), (3.6), (3.31), (3.41), (3.43), (3.44), (3.47), (3.48), (3.8) and (3.49).

And the problem including the machine capacity constraint is represented by the model M3.6(c) given below.

MODEL 3.6(c)

$$\text{Maximize} \sum_{i=1}^{g} \sum_{j=1}^{g} R_{ij}\, x_{ij}$$

Subject to: (3.5), (3.6), (3.31), (3.41), (3.43), (3.47), (3.48), (3.50), (3.8) and (3.49).

For the case when a new production system for cellular manufacturing is to be designed, the grouping problem can be described by model M3.6(d) as given below. In this model, the objective of minimizing the total investment on the machines will allow only the required machines to be assigned to the different groups. Hence, it is not necessary to include the constraint (3.45) into the model.

MODEL M3.6(d)

$$\text{Minimize} \sum_{m=1}^{M} f_m \sum_{j=1}^{g} z_{mj}$$

Subject to: (3.5), (3.6), (3.31), (3.41), (3.43), (3.46), (3.47), (3.48), (3.50),(3.8) and (3.49).

In this case of grouping which is coupled with production system design problem, the other problem scenarios which do not consider restriction on the group disjointedness, will require modifications in the machine capacity constraint. The machine

capacity constraint, for such situation, will be:

$$\sum_{i=1}^{q} t_{im} \sum_{j=1}^{q} x_{ij} \le T_m \sum_{j=1}^{q} z_{mj} \qquad m = 1, \ldots, M.$$

In the other grouping situation, one may consider a production system where few machines are already existing and the changes in the product line require additional machines for cellular production. In this case also, it is reasonable to consider the objective as minimization of extra investment on the machines to be procured. The number of additional machines of type m, denoted as Z_m, can be linked with the variable z_{mj} in the constraint (3.43) as given below.

$$\sum_{j=1}^{q} z_{mj} - Z_m \le |N_m| \qquad m = 1, \ldots, M \quad (3.51)$$

Obviously,

$$Z_m \ge 0 \text{ and integer} \qquad m = 1, \ldots, M. \quad (3.52)$$

In constraint (3.51), $|N_m|$ is assumed to denote the set of existing machines of the type m. Further, the capacities of the new and existing machines of the same type are assumed to be the equal. The generalized grouping problem for this case can be described by the following model.

<u>MODEL M3.6(e)</u>

$$\text{Minimize} \sum_{m=1}^{M} f_m Z_m \qquad (3.53)$$

Subject to: (3.5), (3.6), (3.31), (3.41), (3.46), (3.47), (3.48), (3.50), (3.51), (3.8), (3.49) and (3.52).

3.5.2 Machines Considered Separately

The grouping problem considered in this section identifies each machine separately. For the case discussed in the Section 3.5.1.2, the information on number of copies of each machine type is recorded and capacities of the machines of the same type are assumed to be the same. Whereas in this case, each machine is numbered differently and the information related to its capacity and capability is recorded separately. It can be visualized that in this case, because of the additional input requirement, the size of the problem will be more as compared to that for the problem discussed in the Section 3.5.1.2 where the identical machines are not considered separately for assignment. However, the advantage is derived from the consideration of individual machine capacity which may be different even for identical machines.

It should be noted that the identical machines though numbered differently are to be considered of the same type for the purpose of computing the relative requirement compatibility.

In this case, there are two different ways in which the information regarding the requirements of the parts can be represented:

(a) For each operation of the various parts the machines which are capable of performing them, are listed.

For this kind of requirement, the input information given in the form of machine type and process plan matrix will not suffice because a process plan, in general, does not specify the indices of the machines against the operations and simply lists the types of machine capable of performing

them. However, a route plan which lists the indices of the machines against each operation can be used for this purpose. Thus, in this situation, the necessary information can be provided by giving detail of each of the machines and the route plans.

(b) For each operation of the various parts, the capable machine types are listed.

In this case, the information given in the form as described in the Section 3.5.1 itself will serve the purpose.

The models for these two different situations are given below.

3.5.2.1 INDICES OF MACHINES LISTED AGAINST EACH OPERATION

The expressions for the objective functions and the various constraints are given below.

OBJECTIVE FUNCTION

(i) Maximization of the total relative requirement compatibility

This objective can be expressed as:

$$\text{Maximize} \sum_{\underline{i}=1}^{q} \sum_{\underline{j}=1}^{q} R_{\underline{i},\underline{j}}\, x_{\underline{i},\underline{j}}. \tag{3.54}$$

where $R_{\underline{i},\underline{j}}$ ($\underline{i} \in F_n$; $\underline{j} \in F_{\underline{n}}$ and n, $\underline{n}$ = 1, ..., N) can be calculated using any of the following expressions.

$$\text{(a)}\quad R^{1}_{\underline{i},\underline{j}} = \frac{\displaystyle\sum_{k=1}^{K_n} \mu\left[\sum_{\underline{k}=1}^{K_{\underline{n}}} \lambda\left\{\underline{B}(\underline{i},\ k),\ \underline{B}(\underline{j},\ \underline{k})\right\}\right]}{K_n} \tag{3.55}$$

$$\text{(b)} \quad R^2_{\underline{i},\underline{j}} = \frac{\sum_{k=1}^{K_n} t_{\underline{i},k}\, \mu\left[\sum_{\underline{k}=1}^{K_{\underline{n}}} \lambda\left\{\underline{B}(\underline{i},\ k),\ \underline{B}(\underline{j},\ \underline{k})\right\}\right]}{\sum_{k=1}^{K_n} t_{\underline{i},k}} \tag{3.56}$$

$R^1_{\underline{i},\underline{j}}$ denotes RRC of route plan $\underline{i}$ with respect to $\underline{j}$ and is computed on the basis of number of operations. However, $R^2_{\underline{i},\underline{j}}$ requires processing times of the operations for its computation.

(ii) Minimization of investment on the machines

The expression related to this objective can be written as:

$$\text{Minimize} \sum_{m=1}^{M} f_m \sum_{\underline{m}\in N_m} \sum_{\underline{j}=1}^{\underline{q}} y_{\underline{m},\underline{j}}. \tag{3.57}$$

CONSTRAINTS

(i) Selection of only one route plan for a part

$$\sum_{\underline{i}\in G_n} \sum_{\underline{j}=1}^{\underline{q}} x_{\underline{i},\underline{j}} = 1 \qquad n = 1, \ldots, N \tag{3.58}$$

(ii) Formation of required number of groups

$$\sum_{\underline{j}=1}^{\underline{q}} x_{\underline{j},\underline{j}} = p \tag{3.59}$$

(iii) Assignment of route plans to groups

$$\sum_{\underline{i}=1}^{\underline{q}} x_{\underline{i},\underline{j}} \le (N-p+1)x_{\underline{j},\underline{j}} \qquad \underline{j} = 1, \ldots, \underline{q} \tag{3.60}$$

(iv) Number of parts assigned to a group

$$\sum_{\underline{i}=1}^{\underline{q}} x_{\underline{i},\underline{j}} \le P_U \qquad \underline{j} = 1, \ldots, \underline{q} \tag{3.61}$$

$$\sum_{\underline{i}=1}^{\underline{g}} x_{\underline{i},\underline{j}} \geq P_L \, x_{\underline{j},\underline{j}} \qquad \underline{j} = 1, \ldots, \underline{g} \qquad (3.62)$$

(v) Load on machines

$$\sum_{\underline{i}=1}^{\underline{g}} t_{\underline{i},\underline{m}} \sum_{\underline{j}=1}^{\underline{g}} x_{\underline{i},\underline{j}} \leq T_{\underline{m}} \qquad \forall \, \underline{m} \in N_m; \quad m = 1, \ldots, M \qquad (3.63)$$

(vi) Assignment of a machine to only one group

$$\sum_{\underline{j}=1}^{\underline{g}} y_{\underline{m},\underline{j}} \leq 1 \qquad \forall \, \underline{m} \in N_m; \quad m = 1, \ldots, M \qquad (3.64)$$

(vii) Group disjointedness

The condition on the disjointedness of the groups can be expressed by the constraints similar to those introduced in the Section 3.5.1.1. The constraint that the required machines are available in the corresponding groups, can be represented as:

$$\sum_{\underline{i}=1}^{\underline{g}} a_{\underline{i},\underline{m}} \, x_{\underline{i},\underline{j}} \leq (N-p+1) y_{\underline{m},\underline{j}} \qquad \forall \, \underline{m} \in N_m; \quad m = 1, \ldots, M; \quad \underline{j} = 1, \ldots, \underline{g}. \qquad (3.65)$$

The constraint that only the required machines are assigned, can be written as:

$$\sum_{\underline{i}=1}^{\underline{g}} a_{\underline{i},\underline{m}} \, x_{\underline{i},\underline{j}} \geq y_{\underline{m},\underline{j}} \qquad \underline{m} \in N_m; \quad m = 1, \ldots, M; \quad \underline{j} = 1, \ldots, \underline{g}. \qquad (3.66)$$

(viii) Total number of machines in production system

For the reasons as described in the Section 3.4.1.2, one may like to restrict the total number of machines that can be assigned to different cells. The constraint is expressed as:

$$\sum_{m=1}^{M} \sum_{\underline{m} \in N_m} \sum_{\underline{j}=1}^{\underline{g}} y_{\underline{m},\underline{j}} \leq M_T \tag{3.67}$$

(ix) Total number of machines in a group

The respective constraints on the maximum and the minimum number of parts that can be assigned to a group can be expressed by the constraints (3.68) and (3.69) as given below.

$$\sum_{m=1}^{M} \sum_{\underline{m} \in N_m} y_{\underline{m},\underline{j}} \leq M_U \qquad \underline{j} = 1, \ldots, \underline{g} \tag{3.68}$$

$$\sum_{m=1}^{M} \sum_{\underline{m} \in N_m} y_{\underline{m},\underline{j}} \geq M_L \, x_{\underline{j},\underline{j}} \qquad \underline{j} = 1, \ldots, \underline{g} \tag{3.69}$$

DECISION VARIABLES

$$x_{\underline{i},\underline{j}} = 0 \text{ or } 1 \qquad \underline{i},\underline{j} = 1, \ldots, \underline{g} \tag{3.70}$$

$$y_{\underline{m},\underline{j}} = 0 \text{ or } 1 \qquad \forall\ \underline{m} \in N_m;\ m = 1, \ldots, M;\ \underline{j} = 1, \ldots, \underline{g} \tag{3.71}$$

MODELS

It should be noted that when both the machine capacity and the group disjointedness constraints are cosidered simultaneously, the expression (3.63) representing the constraint on load of machines will not be valid and the constraint (3.65) is not to be considered separately. The two constraints can be represented by the following:

$$\sum_{\underline{i}=1}^{\underline{g}} t_{\underline{i},\underline{m}} \, x_{\underline{i},\underline{j}} \leq T_{\underline{m}} \, y_{\underline{m},\underline{j}} \qquad \forall\ \underline{m} \in N_m;\ m = 1, \ldots, M;\ \underline{j} = 1, \ldots, \underline{g}. \tag{3.72}$$

Similarly, in this case also, the route plan assignment constraint (3.60) can be merged with the constraint (3.61) on the maximum number of parts assigned to a group. The constraint having combined effect, is:

$$\sum_{i=1}^{g} x_{i,j} \geq \left\{ \min \left[(N-p+1),\ P_U\right] \right\} x_{j,j}$$

$$j = 1, \ldots, g. \qquad (3.73)$$

Further, the constraint (3.64) is to be considered only when there is a limit on the number of machines assigned to a cell or to the system as a whole, and also when the group disjointedness is required.

The generalized grouping problem at the operational level, considering the constraints on the machine capacity, the group disjointedness and the number of parts and machines that can be assigned to a group, is represented below by model M3.7(a). For this situation, as discussed earlier, the condition that only required machines are assigned carries no significance, and thus the constraint (3.66) is not included.

MODEL M3.7(a)

$$\text{Maximize} \quad \sum_{i=1}^{g} \sum_{j=1}^{g} R_{i,j}\, x_{i,j}$$

Subject to: (3.58), (3.59), (3.73), (3.62), (3.64), (3.67), (3.68), (3.69), (3.72), (3.70) and (3.71).

The problem of grouping at the planning level becomes a production system design problem where one is interested in determining the configuration of machine cells requiring minimum

investment for disjoint cellular production. For this problem, the objective function and the other constraints are as given in the following model.

MODEL M3.7(b)

$$\text{Minimize} \sum_{m=1}^{M} f_m \sum_{\underline{m} \in N_m} \sum_{\underline{j}=1}^{\underline{q}} y_{\underline{m},\underline{j}}$$

Subject to: (3.58), (3.59), (3.73), (3.62), (3.64), (3.67), (3.68), (3.69), (3.72), (3.70) and (3.71).

It can be seen from the formulations given in the Sections 3.5.1.2 and 3.5.2.1, that the size of the problem for the models discussed in the Section 3.5.2.1 is comparatively more. In addition, the models M3.7(a) and M3.7(b) will not consider any route plans for assignment for which the total processing time requirement for its various operations on any machine is more than the available capacity.

Further, the models M3.7(a) and M3.7(b) may yield a route plan family whose more than one member route plan require the same type of machines but different ones. In this situation, it may be possible to satisfy the total requirement of member route plans by lesser number of machines by making use of the remaining capacity on each such machines. This can only be possible when the plans for their various operations are allowed to share all the machines of the same type. Therefore, it can be seen that the solution to the grouping problem at planning level with disjoint group formation may ask for more machines and thus an extra investment, whereas for the grouping problem at the operational level the objective function may be restricted to assume a lower value. The lower value of the objective function value, of course, can be

improved by assigning to a group only that many machines which satisfy the pooled requirements of the route plans in the corresponding group, and thus by making extra machines, if any, available for assignment to other groups.

3.5.2.2 INDICES OF MACHINE TYPES LISTED AGAINST EACH OPERATION

In this case, though the identical machines are considered separately, the information related to the machine requirements for the operations of the various parts is taken in the usual way as adopted in the Section 3.5.1. For this case, the various expressions for the objective functions and the constraints are the same as given in the Section 3.5.1.2. However, z_{mj} is replaced by $\sum_{\underline{m} \in N_m} y_{\underline{m},j}$ in the expressions where ever it appears.

The objective functions, constraints and the various models are as follows.

OBJECTIVE FUNCTION

(i) <u>Maximization of total relative requirement compatibility</u>:

This objective is the same as shown in the expression (3.27).

(ii) <u>Minimization of investment on the machines</u>

This objective can be expressed as:

$$\text{Minimize} \sum_{m=1}^{M} f_m \sum_{\underline{m} \in N_m} \sum_{j=1}^{q} y_{\underline{m},j}. \tag{3.74}$$

CONSTRAINTS

(i) <u>Selection of only one process plan for a part</u>: (3.5).

(ii) <u>Formation of required number of groups</u>: (3.6).

(iii) <u>Assignment of process plans to groups</u>: (3.23).

(iv) <u>Number of parts assigned to a group</u>

(a) Limit on the maximum number of parts: (3.30) or (3.41).

The constraint (3.41) takes care of the constraint on process plan assignments also.

(b) Limit on the minimum number of parts: (3.31).

(v) Load on machines

The constraint in the absence of the group disjointedness requirement is represented by (3.32) where C_m is given as:

$$C_m = \sum_{\underline{m} \in N_m} T_{\underline{m}} \qquad m = 1, \ldots, M.$$

(vi) Assignment of a machine to only one group

A machine can be assigned to one group at the most. This constraint can be expressed as:

$$\sum_{j=1}^{q} y_{\underline{m},j} \le 1 \qquad \forall\ \underline{m} \in N_m; \quad m = 1, \ldots, M. \tag{3.75}$$

(vii) Group disjointedness

The constraint that all the machines required by a process family are available in the corresponding group, can be expressed as:

$$\sum_{i=1}^{q} a_{im} x_{ij} \le (N-p+1) \sum_{\underline{m} \in N_m} y_{\underline{m},j} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q. \tag{3.76}$$

This constraint along with the constraint on the machine capacity can be written as:

$$\sum_{i=1}^{q} t_{im}\, x_{ij} \le \sum_{\underline{m} \in N_m} T_{\underline{m}}\, y_{\underline{m},j} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q. \tag{3.77}$$

The constraint that only the required machines are assigned, can be written as:

$$\sum_{i=1}^{q} a_{im} x_{ij} \leq \left\{ \frac{1}{|N_m|} \right\} \left\{ \sum_{\underline{m} \in N_m} y_{\underline{m},j} \right\} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q. \quad (3.78)$$

(viii) <u>Total number of machines in production system</u>

$$\sum_{m=1}^{M} \sum_{\underline{m} \in N_m} \sum_{j=1}^{q} y_{\underline{m},j} \leq M_T. \qquad (3.79)$$

(ix) <u>Total number of machines in a group</u>

The limits on the maximum and the minimum number of machines are expressed as:

$$\sum_{m=1}^{M} \sum_{\underline{m} \in N_m} y_{\underline{m},j} \leq M_U \qquad j = 1, \ldots, q, \quad (3.80)$$

and

$$\sum_{m=1}^{M} \sum_{\underline{m} \in N_m} y_{\underline{m},j} \geq M_L \, x_{jj} \qquad j = 1, \ldots, q. \quad (3.81)$$

DECISION VARIABLES

$$x_{ij} = 0 \text{ or } 1 \qquad i, j = 1, \ldots, q$$

$$y_{\underline{m},j} = 0 \text{ or } 1 \qquad \forall\, \underline{m} \in N_m; \quad m = 1, \ldots, M; \quad j = 1, \ldots, q \quad (3.82)$$

MODELS

In this case also, the different combination of the constraints will address different problem situations.

It should be noted that the constraint (3.75) on the machine assignment is to be incorporated only when there are restrictions

on the maximum and/or the minimum number of machines in a group, the total number of machines in the system as a whole, or on the group disjointedness.

The generalized grouping problem at the operational level is as described by the following model.

MODEL M3.8(a)

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij} \, x_{ij}$$

Subject to: (3.5), (3.6), (3.31), (3.41), (3.75), (3.77), (3.79), (3.80), (3.81), (3.8) and (3.82).

The generalized grouping problem at the planning level is given by model M3.8(b) stated below.

MODEL 3.8(b)

$$\text{Minimize} \sum_{m=1}^{M} f_m \sum_{\underline{m} \in N_m} \sum_{j=1}^{q} y_{\underline{m},j}$$

Subject to: (3.5), (3.6), (3.31), (3.41), (3.75), (3.77), (3.79), (3.80), (3.81), (3.8) and (3.82).

In this case also, where the grouping problem is combined with the production system design problem, the machine capacity constraint in the absence of the requirement on group disjointedness will be modified to:

$$\sum_{i=1}^{q} t_{im} \sum_{j=1}^{q} x_{ij} \le \sum_{\underline{m} \in N_m} T_{\underline{m}} \sum_{j=1}^{q} y_{\underline{m},j} \qquad m = 1, \ldots, M.$$

It can be noted that the models M3.8(a) and M3.8(b) are somewhat large in problem size as compared to the respective models M3.6(a) and M3.6(d), but are much smaller as compared to

the respective models M3.7(a) and M3.7(b). Further, the models M3.8(a) and M3.8(b) do not have the shortcomings as mentioned for the models M3.7(a) and M3.7(b). Therefore, the use of the models M3.8(a) and M3.8(b) over the respective models M3.7(a) and M3.7(b) may provide comparatively a better grouping solution.

3.6 SOLUTION METHODOLOGY

The generalized grouping problem formulation proposed by Kusiak (1987) is a modification and extension of p-median problem formulation. Further, it has been shown in the Section 3.4.2 that this formulation is equivalent to GAP formulation of grouping problem proposed by Shtub (1989). The p-median and generalized assignment problems have been reported to be NP-complete (Parker and Rardin (1988)). Further, the constraints incorporated into the basic model (i.e. the p-median problem) does not make the basic constraints non-binding. Thus, the extensions of basic models will also be NP-complete.

Most of these problems are very hard because they consider many parameters, thus finding some reasonably good solution to these problems may not be a simple task. Moreover, in some of the cases, finding a feasible solution itself may be very hard.

The models described in the Section 3.5 are solved using a mathematical programming software LINDO which basically uses branch and bound technique for solving integer programming problems. For large size problems, however, this methodology may not remain efficient. Thus, it may be desirable to find solution of such problems using some efficient methodologies. In the following subsection, a heuristic is described for the problem scenario equivalent to that considered by Kusiak (1987) and Shtub

(1989). In the subsequent subsections, its use for the other problem scenarios with the required modifications are discussed.

3.6.1 Heuristic for the Maximization of the Total Relative Requirement Compatibility

The approach proposed considers the generalized grouping problem where parts have alternative process plans. The problem is to form p groups of process plans such that the sum of the relative requirement compatibility of the various plans in the groups computed with respect to the process plans representing the corresponding groups, is maximized. The problem can be described as:

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij} \, x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j=1}^{q} x_{ij} = 1 \qquad n = 1, \ldots, N$$

$$\sum_{j=1}^{q} x_{jj} = p$$

$$\sum_{i=1}^{q} x_{ij} \le (N-p+1) \, x_{jj} \qquad j = 1, \ldots, q$$

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, q.$$

The above formulation is the same as given for the model M3.5(a) in the Section 3.5.1.1.

The heuristic procedure and the methods for solving certain intermediate problems encountered in the process of using the heuristic are described below. Later, examples are given to illustrate the various steps of the heuristic.

3.6.1.1 PROCEDURE

The various steps of the heuristic are as follows. A flow chart showing the details of the heuristic is given in Figure 3.4.

Selection of p process plans for representing process families

Step 1: For each process plan i, find the sum of the maximum RRC of process plans of each of the other parts computed with respect to plan i and denote it by S_i. Thus,

$$S_i = \sum_{\substack{n=1 \\ n \neq \underline{n}}}^{N} \left\{ \underset{j \in F_n}{\text{maximum}} \left(R_{ji} \right) \right\} \qquad \forall\ i \in F_{\underline{n}};\ \underline{n} = 1, \ldots, N.$$

Step 2: Sequence the process plans in decreasing order of the value of S_i. In case of tie, plans should be arranged in the order of increasing number of machines required.

Step 3: The first p process plans in the sequence obtained in the Step 2 and each belonging to different part will represent p process families. Any plan denoting a family should not have all its machine requirements common to some other process plan that also represents a process family. This should be allowed only when no process plan is available for representing a process family.

Let,

P = set of process plans representing process families

Q = set of remaining process plans ($\notin$ P) whose machine requirements are not completely common to some plan in P

R = set of those parts of which no process plan represents any process family

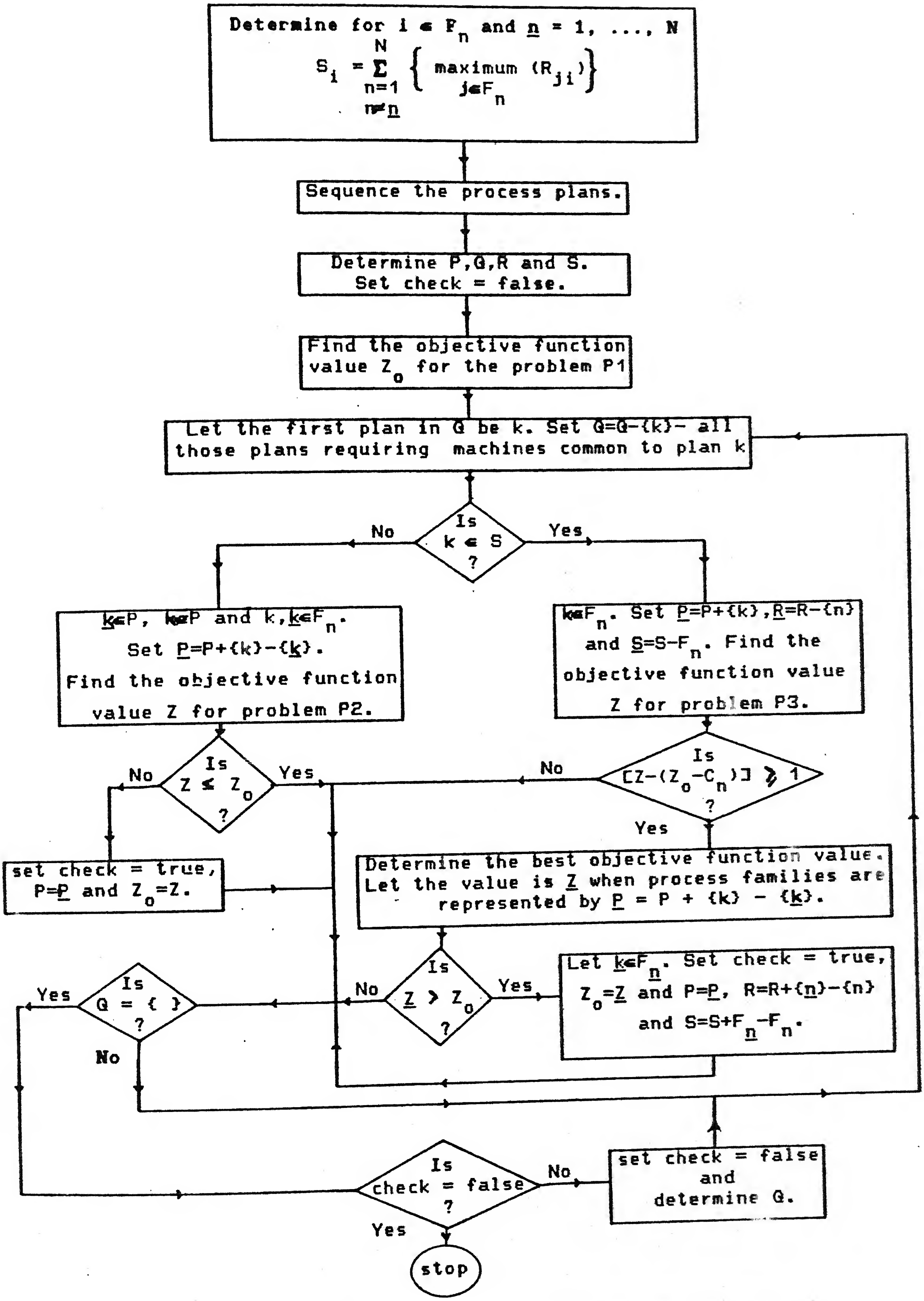


Figure 3.4: Flow Chart for the Heuristic for Determining the Groups While Maximizing the Total of the Similarity Value.

S = set of process plans of parts in R.

The plans in Q are arranged according to the sequence determined in the Step 2.

Set check = false and go to the next step.

Determination of initial grouping solution

Step 4: Solve the problem P3.1, given below.

Problem P3.1

$$\text{Maximize} \sum_{n \in R} \sum_{i \in F_n} \sum_{j \in P} R_{ij} x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in P} x_{ij} = 1 \qquad \forall n \in R$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall i \in F_n; \forall n \in R \text{ and } \forall j \in P$$

Let the objective function value be Z_o. Go to Step 10.

Perturbation for improving the current solution

Step 5: Let the first plan in Q be k. Set Q = Q - {k}. Remove from Q those plans also whose machine requirements are totally common to that of plan k. If $k \in S$, go to Step 7 else to the next step.

Step 6: Let k be the process plan of a part whose some other process plan $\underline{k}$ is in P (i.e. $\underline{k} \in P$, $k \notin P$ and k, $\underline{k} \in F_n$).

Set $\underline{P}$ = P + {k} - {$\underline{k}$}. Solve the problem P3.2, given below.

Problem P3.2

Maximize $\sum_{n \in R} \sum_{i \in F_n} \sum_{j \in \underline{P}} R_{ij} x_{ij}$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in \underline{P}} x_{ij} = 1 \qquad \forall\, n \in R$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall\, n \in R;\ \forall\, i \in F_n \text{ and } \forall\, j \in \underline{P}$$

Let the objective function value for the above problem is Z. If $Z \le Z_o$, then go to Step 10. Otherwise, set check = true, $P = \underline{P}$, $Z_o = Z$ and go to Step 10.

Step 7: Let the process plan k belong to part n (i.e. $k \in F_n$) and the contribution of part n to Z_o be C_n. Set $\underline{P} = P + \{k\}$, $\underline{R} = R - \{n\}$ and $\underline{S} = S - F_n$. Solve the problem P3.3 given below.

Problem P3.3

Maximize $\sum_{n \in \underline{R}} \sum_{i \in F_n} \sum_{j \in \underline{P}} R_{ij} x_{ij}$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in \underline{P}} x_{ij} = 1 \qquad \forall\, n \in \underline{R}$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall\, n \in \underline{R};\ \forall\, i \in F_n \text{ and } \forall\, j \in \underline{P}$$

Let the objective function value for the problem P3.3 be Z. If $[Z-(Z_o-C_n)] \ge 1$, then go to the next step else to Step 10.

Step 8: Find the groups of process plans where one of the process

family is represented by process plan k and rest (p-1) families by (p-1) process plans belonging to P. Let the maximum value of the objective function of these problems is $\underline{Z}$ for $\underline{P} = P + \{k\} - \{\underline{k}\}$. If $\underline{Z} > Z_o$, then go to the next step else to Step 10.

Step 9: Set check = true, $Z_o = \underline{Z}$ and $P = \underline{P}$. Let $\underline{k} \in F_{\underline{n}}$, then set $R = R + \{\underline{n}\} - \{n\}$ and $S = S + F_{\underline{n}} - F_n$ and go to the next step.

<u>Termination of procedure</u>

Step 10: If Q is not empty, go to Step 5. Otherwise, go to the next step.

Step 11: If check = false, then stop. Z_o will be the optimal objective function value.

Otherwise, set check = false, Q as the set of all process plans not in P and having machine requirements different from the plans in P. Go to Step 5.

6.1.2 SOLUTION METHODOLOGY FOR MATHEMATICAL MODELS APPEARING IN THE HEURISTIC

For a better understanding, consider the problem P3.1 given in the Step 4 of the heuristic described in the Section 3.6.1.1. This problem can be decomposed into $|R|$ subproblems. For $n \in R$, these will be:

$$\text{Maximize } Z_n = \sum_{i \in F_n} \sum_{j \in P} R_{ij} \, x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in P} x_{ij} = 1$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall \; i \in F_n \text{ and } \forall \; j \in P.$$

The above problem can be solved in the following manner.

Find $R_{\underline{i},\underline{j}} = \underset{i \in F_n, j \in P}{\text{maximum}} (R_{ij})$. Set $x_{ij} = 0 \; \forall \; i \in F_n$ and $\forall \; j \in P$ except for $x_{\underline{i},\underline{j}}$ which is set equal to one. Thus, $Z_n = R_{\underline{i},\underline{j}}$. The objective function value for the problem P3.1 will be equal to $\sum_{n \in R} Z_n$.

In a similar manner, the mathematical models at the Steps 6, 7 and 8 can also be solved.

3.6.1.3 DISCUSSION ON VARIOUS STEPS OF THE HEURISTIC

In this section, various steps of the heuristic given in the Section 3.6.1.1 along with their relevance are explained. Since the selection of p process plans to represent p process families where each belongs to a different part is a hard exercise for the objective of maximization of the sum of RRC of other plans assigned in the various groups, the effort is made to first select some p process families and then to perturb them in a systematic manner to find improvement in the objective function value.

The Steps 1, 2 and 3 find in somewhat greedy manner the p process plans which will represent the p process families.

Step 4 provides a solution assigning the plans of the other parts to these process families. Steps 5 and onwards seek to perturb the solution to find a better grouping. For this purpose, an additional process plan is chosen to represent a process family. There are two possible situations: one where this process plan may belong to a part whose some other process plan is already representing a process family, and the other in which it belongs to a part of which no process plan represents any process family. In the first case, for the corresponding part the new plan is

allowed to represent the process family instead of the existing one and the solution is determined. This relates to the Step 6 of the heuristic. In the case of improved value of the objective function, new solution is retained and the old one is discarded. While in the other case, solution is determined finding (p+1) groups of process families. This relates to Step 7 of the heuristic. Any improvement in the objective function value (i.e. $(Z-1.0) \geq (Z_o-C_n)$) indicates that bringing this new plan and removing some existing one, for representing p process families, may yield a better solution. Actual test is performed as described in Step 8.

Step 10 terminates the procedure only when the induction of any process plan for representing a process family does not bring improvement in the objective function value.

In Step 3, selection of a process plan i for representing a process family is discouraged if its machine requirements are completely common to that of some other process plan, say plan j, chosen for representing another process family. This is because of the reason that RRC of other plans with respect to j will be at least what is that of these plans with respect to plan i. For the similar reasons, from Q in Step 5 those plans are also rejected whose machine requirements are completely common to process plan k chosen to represent a process family.

3.6.1.4 COMPUTATIONAL COMPLEXITY

The determination of S_i in Step 1 requires computation of maximum of RRCs of alternative process plans of each part with respect to all the other process plans of different parts. The computational requirement of this step, and also of determining

the sequence in Step 2 by arranging the process plans based on the value of S_i, is polynomially bounded (Tremblay and Sorensan, 1988). The computational complexity of Step 3 is linear.

It has been shown in the Section 3.6.1.2 that the process of solving the mathematical models at Steps 4, 6, 7 and 8 will require determination of the maximum of RRC of the alternative process plans of each of the unassigned part computed with respect to plans representing process families. This procedure would also require polynomially bounded computation.

In case of no improvement in the objective function value, the Steps 5 to 10 will be followed for at most (q-p) times, and then the procedure will be terminated. Thus, the computational complexity remains polynomially bounded. In the other case when the Step 5 and onwards bring improvement in the objective function value, the procedure will get terminated in polynomially bounded time as can be seen from further discussion.

Let the minimum positive difference between two RRCs of two different pairs be d. Since the respective upper and lower bounds of the objective function are N and p, Steps 5 to 11 will be followed for at most $(\frac{N-p}{d} + 1)$ times. Since the computational requirements of the other steps are shown to have polynomially bounded complexity, the algorithm will terminate in polynomial time.

3.6.1.5 EXAMPLES

For the illustration of the various steps of the heuristic given in the Section 3.5.1.1, the following examples are taken.

<u>Example 3.2</u>

The data for this example are given in Table 3.5. The RRC of

the various plans computed with respect to each other is presented in Figure 3.5.

The details of the various steps of the heuristic followed to solve the problem, are given below.

Step 1: The sum of the maximum RRC of process plans of the other parts computed with respect to each process plan is:

$\{S_i\}$ = {5.33, 4.66, 4.16, 5.33, 3.83, 5.0, 2.5, 5.0, 0.5, 4.66, 3.16, 3.83, 3.16, 4.16, 1.33}.

Table 3.5: Data For Example 3.2.

$M = 5;\ N = 7;\ p = 2;\ q = 15$

Part (n)	Alternative Process Plan ($i \in F_n$)	Machines 1	2	3	4	5
1	1	1	1	1	1	
	2	1	1	1		
	3		1	1		
2	4	1	1	1	1	
	5	1		1		
3	6			1	1	1
	7				1	1
4	8			1	1	1
	9					1
5	10	1	1	1		
	11			1		
6	12	1		1		
	13			1		
7	14		1	1		
	15				1	

j

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1.00	0.75	0.50	1.00	0.50	0.50	0.25	0.50	0.00	0.75	0.25	0.50	0.25	0.50	0.25
2	1.00	1.00	0.66	0.00	0.66	0.33	0.00	0.33	0.00	1.00	0.33	0.66	0.33	0.66	0.00
3	1.00	1.00	1.00	1.00	0.50	0.50	0.00	0.50	0.00	1.00	0.50	0.50	0.50	1.00	0.00
4	1.00	0.75	0.50	1.00	0.50	0.50	0.25	0.50	0.00	0.50	0.25	0.50	0.25	0.50	0.25
5	1.00	1.00	0.50	1.00	1.00	0.50	0.00	0.50	0.00	1.00	0.50	1.00	0.50	0.50	0.00
6	0.66	0.33	0.33	0.66	0.33	1.00	0.66	1.00	0.33	0.33	0.33	0.33	0.33	0.33	0.33
7	0.50	0.00	0.00	0.50	0.00	1.00	1.00	1.00	0.50	0.00	0.00	0.00	0.00	0.00	0.50
8	0.66	0.33	0.33	0.66	0.33	1.00	0.66	1.00	0.33	0.33	0.33	0.33	0.33	0.33	0.33
9	0.00	0.00	0.00	0.00	0.00	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
10	1.00	1.00	0.66	1.00	0.66	0.33	0.00	0.33	0.00	1.00	0.33	0.66	0.33	0.66	0.00
11	1.00	1.00	1.00	1.00	1.00	1.00	0.00	1.00	0.00	1.00	1.00	1.00	1.00	1.00	0.00
12	1.00	1.00	0.50	1.00	1.00	0.50	0.00	0.50	0.00	1.00	0.50	1.00	0.50	0.50	0.00
13	1.00	1.00	1.00	1.00	1.00	1.00	0.00	1.00	0.00	1.00	1.00	1.00	1.00	1.00	0.00
14	1.00	1.00	1.00	1.00	0.50	0.50	0.00	0.50	0.00	1.00	0.50	0.50	0.50	1.00	0.00
15	1.00	0.00	0.00	1.00	0.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

Figure 3.5: The Relative Requirement Compatibility Matrix $[R_{ij}]$ For Example 3.2.

Step 2: The plans are arranged as shown in the sequence given below considering $\{S_i\}$ and the number of machine types involved with each process plan:

{1, 4, 6, 8, 2, 10, 3, 14, 5, 12, 11, 13, 7, 15, 9}.

Step 3: P = {1, 6}, Q = ϕ, R = {2, 4, 5, 6, 7},

S = {4, 5, 8, 9, 10, 11, 12, 13, 14, 15} and check = false.

In the above, the process plan 4 is not included in P because the machine requirement of plan 4 is the same as that of plan 1. Further, the machine requirement of all the plans not in P are the same as that of either the plan 1 or plan 6, thus Q becomes empty.

Step 4: Process families are: {1, 4, 10, 12, 14} and {6, 8}.

Z_o = 7.00. Step 10 is to be followed.

Step 10: Since Q is empty and check = false, the procedure is terminated.

The grouping solution of {1, 4, 10, 12, 14} and {6, 8} with the objective function value equal to 7.00 is the optimal one.

Example 3.3

The required data for this example are given in Table 3.6. Figure 3.6 represents the RRC of the various process plans computed with respect to each other.

The various steps of the heuristic followed for finding solution to this example are as given below.

Step 1: The sum of the maximum RRC of process plans of the other parts computed with respect to each process plan is shown by the following vector.

$\{S_i\}$ = {3.0,2.0,2.0,3.0,2.5,3.0,2.0,2.5,2.0,2.5}.

Table 3.6: Data For Example 3.3.

$M = 4;\ N = 5;\ p = 2;\ q = 10$

Part (n)	Alternative Process Plan ($i \in F_n$)	Machines			
		1	2	3	4
1	1	1	1		
	2		1		
2	3	1		1	
	4	1	1		
3	5	1			1
	6	1	1		
4	7			1	1
	8		1		1
5	9			1	1
	10	1			1

i \ j	1	2	3	4	5	6	7	8	9	10
1	1.00	0.50	0.50	1.00	0.50	1.00	0.00	0.50	0.00	0.50
2	1.00	0.00	1.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
3	0.50	0.00	1.00	0.50	0.50	0.50	0.50	0.00	0.50	0.50
4	1.00	1.00	0.50	1.00	0.50	1.00	0.00	0.50	0.00	0.50
5	0.50	0.00	0.50	0.50	1.00	0.50	0.50	0.50	0.50	1.00
6	1.00	0.50	0.50	1.00	0.50	1.00	0.00	0.50	0.00	0.50
7	0.00	0.00	0.50	0.00	0.50	0.00	1.00	0.50	1.00	0.50
8	0.50	0.50	0.00	0.50	0.50	0.50	0.50	1.00	0.50	0.50
9	0.00	0.00	0.50	0.00	0.50	0.00	1.00	0.50	1.00	0.50
10	0.50	0.00	0.50	0.50	1.00	0.50	0.50	0.50	0.50	1.00

Figure 3.6: The Relative Requirement Compatibility Matrix $[R_{ij}]$ For Example 3.3.

Step 2: The sequence of the plans arranged given below considering $\{S_i\}$ and the number of machine types involved with each process plan is: {1,4,6,5,8,10,2,3,7,9}.

Step 3: P = {1, 5}, Q = {8, 3, 7, 9}, R = {2, 4, 5}, S = {3, 4, 7, 8, 9, 10} and check = false.

Step 4: Process families are: {1, 4} and {5, 7, 10}. Z_o = 4.50. Step 10 is to be followed.

Step 10: Since Q is not empty, Step 5 is followed.

Step 5: k = 8, Q = {3, 7, 9}. Since k $\in$ S, Step 7 is followed.

Step 7: n = 4, C_n = 0.5, $\underline{P}$ = {1, 5, 8}, $\underline{R}$ = {2, 5}, and $\underline{S}$ = {3, 4, 9, 10}.

Process families are: {1, 4}, {5, 10} and {8}. Z = 5.0.

Since $[Z - (Z_o - C_n)] = 1.0$, Step 8 is followed.

Step 8: The two sets of process families and the corresponding objective function values are as follows.

Set 1: Process families are, {1, 4, 6} and {8, 9} for Z = 4.5.

Set 2: Process families are, {5, 3, 10} and {8, 2} for Z = 4.5.

Since $Z < Z_o$, Step 10 is followed.

Step 10: Since Q is not empty, Step 5 is followed.

Step 5: k = 3 and Q = {7, 9}. Since k $\in$ S, Step 7 is followed.

Step 7: Again no improvement is found, Step 5 is finally followed.

Step 5: k = 7 and Q = ϕ. Since k $\in$ S, Step 7 is followed.

Step 7: n = 4, C_n = 0.5, $\underline{P}$ = {1, 5, 7}, $\underline{R}$ = {2, 5} and

$\underline{S}$ = {3, 4, 9, 10}.

Process families are: {1, 2}, {5} and {7, 9}. Z = 5.

Since $[Z - (Z_o - c_n)] = 1$, Step 8 is followed.

Step 8: The two set of solutions are:

Set 1: Process families are, {1, 4, 6} and {7, 9}.

Z = 5.0.

Set 2: Process families are, {5, 1, 3} and {7, 9}.

Z = 4.0.

Therefore, $\underline{Z}$ = 5.0 for $\underline{P}$ = {1, 7}. Since $\underline{Z} > Z_o$, Step 9 is followed.

Step 9: Set check = true, Z_o = 5.0, P = {1, 7}. Since $\underline{k}$ = 5 and $\underline{n}$ = 3, R = {2, 3, 5} and S = {3, 4, 5, 6, 9, 10}. Step 10 is now to be followed.

Step 10: Since Q is empty, Step 11 will be followed.

Step 11: Since check = true, then P = {1, 7}, Q = {3, 5, 8, 10}, R = {2, 3, 5} and S = {3, 4, 5, 6, 9, 10}. Step 5 will now be followed.

Further steps do not bring improvement. Therefore, the grouping solution of {1, 4, 6} and {7, 9} with the objective function value equal to 5.0 will be the optimal one.

3.6.2 Generalizations of the Heuristic

In the following subsections, some problems are discussed which can be solved using the heuristic proposed in the Section 3.6.1.1 after making certain modifications. The various problems in terms of the additional constraints are given in the following subsections along with the necessary modifications to be incorporated into the heuristic.

3.6.2.1 CONSTRAINT ON THE MAXIMUM NUMBER OF PARTS IN A GROUP

In this case, the specified number of groups of the process plans are to be determined such that the number of process plans assigned in each of the groups is less than or equal to some prespecified number P_U. The problem can be expressed as given below.

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij} \, x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j=1}^{q} x_{ij} = 1 \qquad n = 1, \ldots, N$$

$$\sum_{j=1}^{q} x_{jj} = p$$

$$\sum_{i=1}^{q} x_{ij} = \left\{\text{minimum } [(N-p+1), P_U]\right\} x_{jj} \qquad j = 1, \ldots, q$$

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, q$$

The above formulation is the same as given for the model M3.5(c) in the Section 3.5.1.1. For this model, the necessary modifications required to be incorporated into the heuristic are as follows.

If $P_U \geq (N-p+1)$, then no modification is required and the heuristic can be used as it is. However, if $P_U < (N-p+1)$, then the constraint

$$\sum_{n \in R} \sum_{i \in F_n} x_{ij} \leq P_U - 1, \qquad \forall \, j \in P,$$

is to be added to the problems P3.1 and P3.2, and the constraint

$$\sum_{n \in \underline{R}} \sum_{i \in F_n} x_{ij} \le P_U - 1, \qquad \forall\ j \in \underline{P}$$

to the problem P3.3.

Similarly, suitable constraints are to be incorporated for the problems to be solved in the Step 8. In the right hand side of the above constraints, the number is P_U-1. It is because of the reason that to each process family one process plan which represents it, is already assigned.

After the incorporation of the constraint described above, the problems at the Steps 4, 6, 7 and 8 assume the structure of an unbalanced transportation problem which can be solved efficiently and easily.

The other steps of the heuristic will remain the same.

3.6.2.2 CONSTRAINT ON THE MINIMUM NUMBER OF PARTS IN A GROUP

In this case, the required number of groups are formed ensuring that to each of these groups some minimum prespecified number of plans P_L are assigned. This problem is the same as described by the model M3.5(b) and is expressed as:

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij}\, x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j=1}^{q} x_{ij} = 1 \qquad n = 1, \ldots, N$$

$$\sum_{j=1}^{q} x_{jj} = p$$

$$P_L\, x_{jj} \le \sum_{i=1}^{q} x_{ij} \le (N-p+1)\, x_{jj} \qquad j = 1, \ldots, q$$

$$x_{ij} = 0 \text{ or } 1 \qquad\qquad i,j = 1, \ldots, q.$$

Corresponding to the above formulation the problems P3.1 in the Step 4 of the heuristic will be as described by the following model.

<u>MODEL M3.9(a)</u>

$$\text{Maximize } \sum_{n \in R} \sum_{i \in F_n} \sum_{j \in P} R_{ij}\, x_{ij} \tag{3.83}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in P} x_{ij} = 1 \qquad \forall\, n \in R \tag{3.84}$$

$$\sum_{n \in R} \sum_{i \in F_n} x_{ij} \geq P_L - 1 \qquad \forall\, j \in P \tag{3.85}$$

$$\sum_{n \in R} \sum_{i \in F_n} x_{ij} \leq \left\{N-[(p-1)P_L+1]\right\} \qquad \forall\, j \in P \tag{3.86}$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall\, n \in R; \; \forall\, i \in F_n \text{ and } \forall\, j \in P. \tag{3.87}$$

The constraint (3.85) takes care of the restriction on minimum number of parts to be assigned to each group. The constraint (3.86) puts an upper bound on the assignments of plans to process families. Since to a process family j one plan is already assigned and to the remaining (p - 1) families at least P_L plans are to be assigned, the maximum number of process plans left to be assigned to plan j, therefore, will be equal to $[N-\{(p-1)P_L+1\}]$. Because of this reason, the number after the inequality in the constraint (86) is $[N-\{(p-1)P_L+1\}]$.

The constraints (3.85) and (3.86) can be written as:

$$\sum_{n\in R} \sum_{i\in F_n} x_{ij} - u_j = P_L - 1 \qquad \forall\ j \in P \qquad (3.88)$$

and
$$\sum_{n\in R} \sum_{i\in F_n} x_{ij} + v_j = N - [(p-1)P_L + 1] \qquad \forall\ j \in P \qquad (3.89)$$

where $u_j,\ v_j \geq 0$ and integer $\qquad \forall\ j \in P. \qquad (3.90)$

From the equations (3.88) and (3.89), the following is obtained.

$$u_j + v_j = N - pP_L \qquad \forall\ j \in p \qquad (3.91)$$

Further, from the equation (3.84)

$$\sum_{n\in R} \sum_{i\in F_n} \sum_{j\in P} x_{ij} = (N - p), \qquad (3.92)$$

and from the equation (3.88)

$$\sum_{n\in R} \sum_{i\in F_n} \sum_{j\in P} x_{ij} - \sum_{j\in P} u_j = p(P_L - 1). \qquad (3.93)$$

Therefore, from the equations (3.92) and (3.93),

$$\sum_{j\in P} u_j = N - pP_L. \qquad (3.94)$$

In view of the above expressions and simplifications, the grouping problem represented by the model M3.9(a) can also be expressed by the following model.

<u>MODEL M3.9(b)</u>

Maximize $\sum_{n\in R} \sum_{i\in F_n} \sum_{j\in P} R_{ij}\, x_{ij}$

Subject to: (3.84), (3.89), (3.91), (3.94), (3.87) and (3.90).

The structure of the above problem is as that of a balanced

transportation problem. The transportation matrix corresponding to the above formulation is given in Figure 3.7.

In the Figure 3.6, j_1 to j_p corresponds to the indices representing p process families. Let i, k ϵ R, and k be the last part in R. The constraint (3.84) is represented together by the columns of Block A and the rows of Block III. The rows of Block I represent the constraint (3.89). The constraint (3.91) is represented by the columns of Block B, while the row of Block II represents the constraint (3.94).

In the model M3.9(a), in fact, the constraint (3.86) is not required. However, it is used to make the problem more tighter. As can be seen from the preceding paragraph that the incorporation of this constraint provides the problem the structure of a transportation problem which can easily be solved. In case, the constraint (3.86) is not included, the formulation of the problem (the model M3.9(a)) after certain simplifications will be as given in the following model.

<u>MODEL M3.9(c)</u>

$$\text{Maximize} \sum_{n \in R} \sum_{i \in F_n} \sum_{j \in P} R_{ij} \, x_{ij} \qquad (3.83)$$

Subject to:(3.84), (3.88), (3.94), (3.87) and (3.90).

The above formulation possesses the property of total unimodularity. Therefore, the solution to the LP relaxation of the problem will itself be optimal to the original problem.

Either of the methods described above can suitably be used for the modified problem P3.2 and P3.3 (in the Steps 6 and 7 of the heuristic) which includes the requirement on the minimum

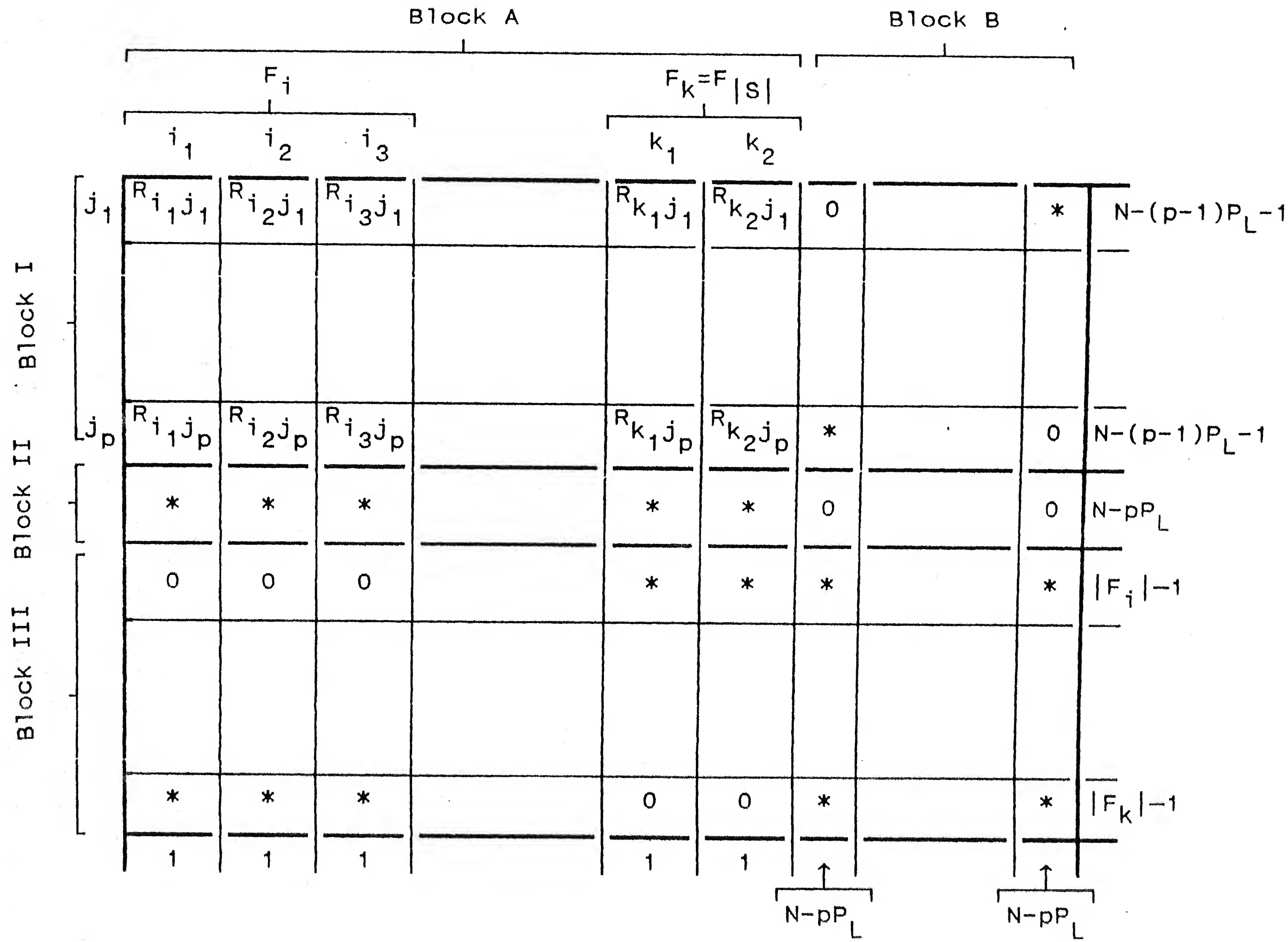


Figure 3.7: Transportation Matrix Corresponding to the Problem P3.1 with Restriction on the Minimum Number of Parts in a Group (* in a cell represents a value of $-\infty$).

number of parts in a group. The formulations of the problem P3.3 obtained after modifications will be similar to that given by the models M3.9(b) or M3.9(c) with P replaced by $\underline{P}$, R by $\underline{R}$ and p by (p+1).

It should be noted that the modifications made above at Step 7 will require $N \geq (p+1)P_L$. In case this condition does not hold, instead of solving the problem P3.3 in Step 7 of the heuristic, Step 8 should immediately be followed.

3.6.2.3 CONSTRAINTS ON THE MINIMUM AND THE MAXIMUM NUMBER OF PARTS IN A GROUP

In this case, the formulation of the problem is the same as given in the model M3.5(d) and is expressed as:

$$\text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij} \, x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j=1}^{q} x_{ij} = 1 \qquad n = 1, \ldots, N$$

$$\sum_{j=1}^{q} x_{jj} = p$$

$$P_L \, x_{jj} \leq \sum_{i=1}^{q} x_{ij} \leq \left\{\text{minimum } [(N-p+1), P_U]\right\} x_{jj} \qquad j = 1, \ldots, q$$

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, q.$$

In the above formulation, if $P_U \geq (N-p+1)$, then the constraint on the number of parts in a group will be non binding and the problem will be equivalent to that represented by the model M3.5(b). For a case of this kind, the solution methodology

discussed in the Section 3.6.2.2 can be used.

In case when $P_U < (N-p+1)$, the problem P3.1 at Step 4 of the heuristic will be as described by the following model.

MODEL M3.9(d)

$$\text{Maximize} \sum_{n \in R} \sum_{i \in F_n} \sum_{j \in P} R_{ij}\, x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in P} x_{ij} = 1 \qquad \forall\ n \in R$$

$$\sum_{n \in R} \sum_{i \in F_n} x_{ij} \geq P_L - 1 \qquad \forall\ j \in P$$

$$\sum_{n \in R} \sum_{i \in F_n} x_{ij} \leq P_U - 1 \qquad \forall\ j \in P$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall\ n \in R;\ \forall\ i \in F_n \text{ and } \forall\ j \in P.$$

Certain constraints in the above model can be simplified in a manner used for transforming model M3.9(a) to M3.9(b). The model after simplification will be as given below.

MODEL M3.9(e)

$$\text{Maximize} \sum_{n \in R} \sum_{i \in F_n} \sum_{j \in P} R_{ij}\, x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in P} x_{ij} = 1 \qquad \forall\ n \in R$$

$$\sum_{n \in R} \sum_{i \in F_n} x_{ij} + v_j = P_U - 1 \qquad \forall\ j \in P$$

$$v_j + u_j = P_U - P_L \qquad \forall\ j \in P$$

$$\sum_{j \in P} u_j = N - pP_L$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall\ n \in R;\ \forall\ i \in F_n \text{ and } \forall\ j \in P$$

$$u_j,\ v_j \geq 0 \text{ and integer} \qquad \forall\ j \in P.$$

The above model also has the basic structure of a transportation problem. Thus, this model can also be represented in the form of a transportation matrix similar to that shown in the Figure 3.7. However, the supply values for the rows of Block I will be (P_U-1) instead of $[N-(p-1)P_L-1]$, and the requirement values for the columns of Block B will (P_U-P_L) instead of $(N-pP_L)$.

In a similar manner, the problems P3.2 and P3.3 to be solved at Steps 6 and 7 of the heuristic can be modified to suit the requirements of the present grouping problem. The formulation for the problem P3.3 will be similar to that in the model M3.9(e) with P replaced by $\underline{P}$, R by $\underline{R}$ and p by (p+1) in this model.

In this case also, the above modification will require $N \geq (p + 1)P_L$. If this restriction is not satisfied, the problem P3.3 should not be solved at Step 7, and Step 8 should be followed immediately.

3.6.3 Lagrangian Relaxation Method

As mentioned in the beginning of this section, the mathematical models reported and developed in the Sections 3.2, 3.4 and 3.5 are NP-Complete. Such integer programming problems can be solved efficiently if the complicating constraints (i.e.

the constraints that tend to turn the problem as NP-complete such as the constraints on assignment of process/route plans, disjointedness of groups, etc.) are relaxed. One of the most popular relaxation is based on **Lagrangian Multipliers**. The advantages and methodology of Lagrangian relaxation for solving various NP-complete problems have been discussed in length by Fisher (1981). The approach has also been described in detail by Geoffrion (1974) and Shapiro (1979). To solve the problems based on Lagrangian relaxation, subgradient algorithm proposed by Held et al (1974) can be used.

In the following subsections, two different grouping problems are considered and the formulations based on the Lagrangian relaxation are given. The first problem relates to the determination of part families considering the limits on the maximum and the minimum number of parts. The second problem relates to the simultaneous determination of groups of parts and machines considering (i) the limits on the maximum and the minimum number of parts, (ii) the limits on the maximum number of machines that can be assigned to a group, (iii) the limit on the total number of machines of each type, (iv) the restriction on group disjointedness and (v) the load on machines.

3.6.3.1 DETERMINATION OF PART FAMILY ONLY

The problem of determination of part family with the related constraints is the same as given in the model M3.5(d). In case when each part has only one process plan (i.e. q = N), and when there are no limits on the number of parts in a group, the problem can be represented by a simplified version of the model M3.5(a) (by taking the index of the parts and the process plan to be the

same) and is written as:

MODEL M3.11

$$\text{Maximize} \quad \sum_{i=1}^{N} \sum_{j=1}^{N} R_{ij}\, x_{ij} \tag{3.95}$$

Subject to:

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1, \ldots, N \tag{3.96}$$

$$\sum_{j=1}^{N} x_{jj} = p \tag{3.97}$$

$$\sum_{i=1}^{N} x_{ij} \leq (N-p+1)\, x_{jj} \qquad j = 1, \ldots, N \tag{3.98}$$

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, N. \tag{3.99}$$

The above model is the same as proposed by Kusiak (1983). From his model (Model M3.12) described below, it can be noted that the constraint (3.98) of model M3.11 is replaced by the constraint (3.101) and the measure of commonality R_{ij} of model M3.11 by d_{ij} in the objective function expression. Further, the objective of minimizing the sum of the distances (dissimilarity) defines as d_{ij} (total number of machines minus simple matching similarity value) for a pair of parts i and j is equivalent to maximizing the sum of the similarity between parts. Thus, the two models are equivalent.

MODEL M3.12

$$\text{Minimize} \quad \sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij}\, x_{ij} \tag{3.100}$$

Subject to:

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1, \ldots, N$$

$$\sum_{j=1}^{N} x_{jj} = p$$

$$x_{ij} \le x_{jj} \qquad i, j = 1, \ldots, N \qquad (3.101)$$

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, N.$$

Kusiak proposed a Lagrangian relaxation of the model M3.12 where the constraint (3.96) is dualized. The resultant problem after relaxation is as given below.

$$(LR_1) \qquad \text{Maximize } Z = \sum_{i=1}^{N} \sum_{j=1}^{N} (d_{ij}-u_i)x_{ij} + \sum_{i=1}^{N} u_i$$

Subject to: (3.97), (3.101) and (3.99),

where the dual variables u_is associated with the constraint (3.96) are unrestricted.

It can be recalled from the discussions made in the preceding paragraphs and also from the elaborations given in the Section 3.4.1 that the constraint (3.98) and (3.101) are equivalent. Since the constraint (3.101) is similar to a Knapsack constraint, the problem LR_1 will be NP-complete. Thus, finding the solution

of this problem even for the known values of u_i^s may not be a simple task. However, if the constraint (3.101) is dualized then the related problem will be:

$$(LR_2) \qquad \text{Maximize } Z = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \neq i}}^{N} (d_{ij}+u_{ij})x_{ij} + \sum_{i=1}^{N} d_{ii}x_{ii}$$

Subject to:

(3.96), (3.97), (3.99) and

$u_{ij} \geq 0 \qquad\qquad i, j = 1, \ldots, N.$

For some known values of u_{ij}^s, the above model will have the property of total unimodularity and thus LP relaxation of this problem will provide an optimal solution even to this problem LR_2. Further, the relaxation of the above problem requires use of N^2 number of u_{ij} variables. If the constraint (3.98) is used instead of the constraint (3.101), then the relaxed formulation is as given below.

$$(LR_3) \qquad \text{Maximize } Z = \sum_{i=1}^{N} \sum_{\substack{j=i \\ j \neq i}}^{N} (d_{ij}+u_j)x_{ij} + \sum_{j=1}^{N} [d_{jj}-(N-p)u_j]x_{jj}$$

Subject to:

(3.96), (3.97), (3.99) and

$u_j \geq 0 \qquad\qquad j = 1, \ldots, N$

The formulation in LR_3 requires only N number of u_j variables.

Thus, for solving the model M3.5(d) which considers the

problem of determination of groups of process plans including the constraints on the number of parts in a group, the complicating constraints would be relaxed in a manner similar to that suggested in the above paragraphs. For the convenience, the formulation of the process family formation is reproduced here.

$$(P_1) \qquad \text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij}\, x_{ij}$$

Subject to:

$$\sum_{i \in F_n} \sum_{j=1}^{q} x_{ij} = 1 \qquad n = 1, \ldots, N \qquad (3.102)$$

$$\sum_{j=1}^{q} x_{jj} = p \qquad (3.103)$$

$$\sum_{i=1}^{q} x_{ij} \leq P_U\, x_{jj} \qquad j = 1, \ldots, q \qquad (3.104)$$

$$\sum_{i=1}^{q} x_{ij} \geq P_L\, x_{jj} \qquad j = 1, \ldots, q \qquad (3.105)$$

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, q. \qquad (3.106)$$

In place of P_U in the right hand side of the constraint (3.104), in fact, (min $[P_U, (N-p+1)]$) should have been. But assuming that upper limit on the number of parts is a meaningful one (i.e. $P_U > (N-p+1)$), only P_U is included.

On dualizing the complicating constraints (3.104) and (3.105), the relaxed problem will be:

$$(LR_4) \quad Z_r = \text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij}\, x_{ij} + \sum_{j=1}^{q} u_j \left(\sum_{i=1}^{q} x_{ij} - P_U\, x_{jj} \right)$$

$$+ \sum_{j=1}^{q} v_j \left(P_L\, x_{jj} - \sum_{i=1}^{q} x_{ij}\right)$$

$$= \text{Maximize} \sum_{i=1}^{q} \sum_{\substack{j=1 \\ j \neq i}}^{q} (R_{ij} + u_j - v_j)\, x_{ij} \qquad (3.107)$$

$$+ \sum_{j=1}^{q} [R_{jj} - u_j(P_U - 1) + v_j(P_L - 1)]\, x_{jj}$$

Subject to:

(3.102), (3.103), (3.106) and

$u_j,\ v_j \geq 0$ and integer $\quad j = 1, \ldots, q. \qquad (3.108)$

The solution to the problem LR_4 provides an upper bound for the objective function value of the problem P_1, hence it will be desirable to minimize Z_r. The best choice of u and v can be found from the following.

The corresponding dual problem will be:

$(D_1) \quad Z_D = \underset{u,\ v}{\text{Minimize}}\ Z_r$

Subject to: (3.102), (3.103), (3.106) and (3.108).

It should be noted that the model LR_4 for some known values of u_j's and v_j's, will have the property of total unimodularity, and thus solving it as an LP problem one can get the optimal solution for LR_4. To solve the problem D_1, an algorithm based on the framework of the subgradient optimization procedure is used. The various steps of the algorithm are given below.

Algorithm

Step 0: Set $k = 1$, $u_j^k = 0 \; \forall \; j$, $v_j^k = 0 \; \forall \; j$, $LB^o = 0.0$, $UB^o = N$ and $\lambda^k = \lambda^o$, where

u_j^k : value of dual variable u_j at iteration k

v_j^k : value of dual variable v_j at iteration k

LB^o : initial lower bound

UB^o : initial upper bound

λ^k : value of the scalar at iteration k
($0.0 \le \lambda^o \le 2.0$).

Step 1: Solve the problem LR_4 for the current set of multipliers u^k and v^k. Let x^k be the optimal solution and the optimal objective function value be $Z_r(u^k, v^k)$.

Step 2: Update the upper bound:

UB^k = minimum $[UB^{k-1}, Z_r(u^k, v^k)]$.

Step 3: If x^k is feasible to P_1, then set $Z_P^k = Z_P(x^k)$ and go to the next step. Otherwise, modify x^k to get a feasible solution for the problem P_1. Let the modified solution be represented by $\underline{x}^k$. Set $Z_P^k = Z_P(\underline{x}^k)$ and go to the next step.

Step 4: Update the lower bound:

LB^k = maximum $[LB^{k-1}, Z_p(x^k)]$.

Step 5: Determine the value of λ^k. If $\lambda^k < \epsilon_1$, stop.

Step 6: If $(UB^k - LB^k)/LB^k < \epsilon_2$, then stop. Otherwise, go to the next step.

Step 7: Compute the subgradients:

$$g_j^k = \sum_{i=1}^{q} x_{ij}^k - P_U . x_{jj}^k \qquad j = 1, \ldots, q$$

$$h_j^k = \sum_{i=1}^{q} x_{ij}^k - P_L . x_{jj}^k \qquad j = 1, \ldots, q$$

Step 8: Compute the step size:

$$f^k = \frac{\lambda^k \left[Z_P^k - LB^k \right]}{\sum_{j=1}^{q} \left[\left\{ g_j^k \right\}^2 + \left\{ h_j^k \right\}^2 \right]}$$

Step 9: Update the Lagrange multipliers:

$$u_j^{k+1} = \text{maximum } (0,\ u_j^k + f^k g_j^k) \qquad j = 1, \ldots, q$$

$$v_j^{k+1} = \text{maximum } (o,\ v_j^k + f^k h_j^k) \qquad j = 1, \ldots, q$$

Set k = k + 1 and go to Step 1.

In the Steps 5 and 6, the two numbers ϵ_1 and e_2 characterizing the convergence define the termination of the algorithm. The values of these numbers are to be specified by the user. The details on the determination of the modified values of λ^k (at the Step 5) and that on the stopping criterion (at the Steps 5 and 6) are given by Held et al (1974) and Geoffrion (1974).

The solution to the dual problem D_1 at the Step 1 can be determined by solving the model LR_4 as an LP. Since the simplex method for solving LP problem is not polynomially bounded and the

solution to the dual problem provides an upper bound to the primal problem P1, to reduce the computational complexity a polynomially bounded procedure can be used for solving the dual problem but at the cost of a higher objective function value.

Given below is a heuristic for solving the problem LR_4. The methodology discussed thereafter can be used at step 3 for finding a feasible solution to the problem P_1 at step 3.

Method for Solving the Dual Problem

For the purpose of solving the problem LR_4 at step 1, a method based on the concepts of the heuristic given in the Section 3.6.1.1, is proposed. Various steps are as follows.

(i) For each $j \in F_n$ and $n = 1, \ldots, N$ find

$$S_j = \sum_{\substack{\underline{n}=1 \\ \underline{n}\neq n}} \left\{ \underset{i \in F_{\underline{n}}}{\text{maximum}} \; (R_{ij}+u_j-v_j) \right\} + [R_{jj}-u_j(P_U-1) + v_j(P_L-1)].$$

(ii) Sequence the plans ($j = 1, \ldots, q$) in the order of decreasing value of S_j^s determined in (i) above.

(iii) The first p process plans in the sequence and each belonging to different part are chosen to represent the process families. Let P denote the set of indices of process plans that represent p process families, and R denote those parts whose no process plans represent any process family. Now for each part $n \in R$, the index of the process plan $\underline{i} \in F_n$ and that of the group $\underline{j} \in P$ to which plan $\underline{i}$ will be assigned, can be determined as follows.

$$R_{\underline{i},\underline{j}} = \underset{i \in F_n; j \in P}{\text{maximum}} (R_{ij}+u_j-v_j) \qquad \forall\ n \in R$$

<u>Method for Finding Feasible Solution for the Primal Problem</u>

The need for finding a feasible solution to the primal problem P_1 arises when the solution obtained for the dual problem at the Step 1 is not primal feasible. The feasible solution can be obtained in a manner given below.

Denote the set of indices j representing process families, i.e. of those j's for which $x^k_{jj} = 1$, as P. It should be noted that the total number of elements in the set P will be p and equal to the required number of process families. Let R denote the set of those parts of which no process plan represents any process family. The problem of assignment of process plans of parts ($n \in R$) to the groups ($j \in P$) is the same as solving the model M3.9(d) discussed in the Section 3.6.2.3.

3.6.3.2. SIMULTANEOUS DETERMINATION OF GROUPS OF PARTS AND MACHINES

The problem of simultaneous determination of groups of parts and machines with the constraints on the number of the maximum and the minimum number of parts in a group, the maximum number of machines in a group, the number of machines available of each type, the load on the machines and the disjointedness of the groups, can be written as:

$$(P_2) \qquad \text{Maximize} \sum_{i=1}^{q} \sum_{j=1}^{q} R_{ij}\ x_{ij}$$

Subject to:

(3.102), (3.103), (3.104), (3.105) and

$$\sum_{m=1}^{M} Z_{mj} \le M_U \qquad j = 1, \ldots, q \qquad (3.109)$$

$$\sum_{j=1}^{q} Z_{mj} \le |N_m| \qquad m = 1, \ldots, M \qquad (3.110)$$

$$\sum_{i=1}^{q} t_{im} x_{ij} \le T_m Z_{mj} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q \qquad (3.111)$$

$$x_{ij} = 0 \text{ or } 1 \qquad i, j = 1, \ldots, q.$$

$$Z_{mj} \ge 0 \text{ and integer} \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q. \qquad (3.112)$$

On dualizing the constraints (3.104), (3.105) and (3.111), the relaxed problem is:

$$(LR_4) \quad Z_r = \text{Maximize} \sum_{i=1}^{q} \sum_{\substack{j=1 \\ j \ne i}}^{q} (R_{ij} + u_j - v_j + \sum_{m=1}^{M} w_{mj} t_{im}) x_{ij} +$$

$$\sum_{j=1}^{q} [R_{jj} - u_j(P_U - 1) + v_j(P_L - 1) + \sum_{m=1}^{M} w_{mj} t_{jm}] x_{jj}$$

$$- \sum_{m=1}^{M} \sum_{j=1}^{q} w_{mj} T_m Z_{mj}$$

Subject to:

(3.102), (3.103), (3.109), (3.110), (3.106), (3.108),

$$(3.112) \text{ and } w_{mj} \geq 0 \qquad m = 1, \ldots, M; \quad j = 1, \ldots, q \tag{3.113}$$

The problem can be decomposed into two subproblems as given below.

$$(SP_1)\ Z_{r1} = \text{Maximize} \sum_{i=1}^{q} \sum_{\substack{j=1 \\ j \neq i}}^{q} (R_{ij} + u_j - v_j + \sum_{m=1}^{M} w_{mj}\, t_{im})\, x_{ij} + \sum_{j=1}^{q} [R_{jj} - u_j(P_U - 1) + v_j(P_L - 1) + \sum_{m=1}^{M} w_{mj}\, t_{jm}]\, x_{jj}$$

Subject to: (3.102), (3.103), (3.106), (3.108) and (3.113).

$$(SP_2)\ Z_{r2} = \text{Maximize} \left[- \sum_{m=1}^{M} \sum_{j=1}^{q} w_{mj}\, T_m\, Z_{mj} \right] \tag{3.114}$$

Subject to: (3.109), (3.110), (3.112) and (3.113).

It can be observed that the problem SP_1 is similar to the problem LR_4, thus the solution methodology discussed before for solving the problem LR_4 can be used for SP_1.

Further, in the problem SP_2, T_m and w_{mj} are nonnegative. Thus, for the objective as described by the expression (3.114), the variable Z_{mj} will be always equal to zero for all m and for all j. Moreover, for the determination of a feasible solution to P_1 from the dual solution, one would be required to solve, in fact, the problem P_1 itself. Thus, for the case of simultaneous grouping Lagrangian relaxation will not be advantageous and it will be better to solve the problem P_1 itself.

3.6.4 Some Comments

Following are the few remarks and suggestions regarding the use of the heuristic described in the Section 3.6.1.1, and also regarding the approach for grouping of parts and machines.

3.6.4.1 INCORPORATION OF OTHER COMMONALITY MEASURES IN THE HEURISTIC

The heuristic in the Section 3.6.1.1 is described using RRC as a measure of commonality. However, the other commonality measures can as well be used in the procedure. At the Step 7, the condition for checking the improvement in the objective function value will be $\{Z-(Z_o+C_n)\} \geq s_{kk}$ $(k \epsilon F_n)$ where s_{kk} is the value of commonality measure for the k^{th} process plan determined with itself.

3.6.4.2 GROUPING APPROACH

The simultaneous determination of part family and machine cell requires a large size problem to be solved. Further, the computational requirements may grow at increasing rate if additional features, such as machine capacity, cell size restrictions, etc., are to be incorporated. Thus, to reduce the computational burden, the groups of parts and machines can be determined in a hierarchical manner.

In the hierarchical approach proposed by Gunasingh and Lashkari (1989a) the machines cells are determined first and then to these machine cells parts are assigned. The problem of machine cell determination is formulated as a p-median problem, whereas that of the assignment of parts to the machine cells as a generalized assignment problem.

It should be noted that the approach suggested by Gunasingh

considered in the present work. The environment considered by them deals with the grouping problem where each operation of a part requires only one tool type and no two operations of a part require the same tool type. An operation is performed on any of the machines which supports the required tool type. The present grouping environment is a general one, in that it allows the processing of an operation using more than one tool type on the same or different machine types. The compatibility measure used in their models cannot accommodate such generalities. Thus, the hierarchical approach of Gunasingh and Lashkari (1989a) which determines the machine cells first cannot be adopted in the present context.

The problem of grouping in a general situation as mentioned above, however, is to be solved by first determining the groups of process plans (i.e. part family) and then assigning machines to such families. The formulation of the problem for the determination of groups of process plans considering the limits on the number of parts in the groups is and extended version of the p-median problem which can be solved using the heuristic proposed in the Section 3.6.1.1 or some mathematical programming technique as described in the Section 3.6.3.1. After the part family has been determined, machine cells can be obtained by assigning machines to the various part families considering various related factors such as the number of machines available of each type, the limit on the size of each machine cell, the total number of machines in the production system, etc..

The machine assignment problem can be described by the following model using two new notations A_{km} and x_{km} defined thereafter.

$$\text{Maximize} \sum_{k=1}^{p} \sum_{m=1}^{M} A_{km}\, x_{km}$$

Subject to:

$$\sum_{k=1}^{p} x_{km} \leq |N_m| \qquad m = 1, \ldots, M \qquad (3.115)$$

$$\sum_{m=1}^{M} x_{km} \geq M_L \qquad k = 1, \ldots, p \qquad (3.116)$$

$$\sum_{m=1}^{M} x_{km} \leq M_U \qquad k = 1, \ldots, p \qquad (3.117)$$

$$\sum_{m=1}^{M} \sum_{k=1}^{p} x_{km} \leq M_T \qquad (3.118)$$

where,

A_{km} = the sum of absolute requirement compatibility of parts in group k for their selected process plans computed with respect to machine m

$$x_{km} = \begin{cases} 1 & \text{if to group k machine m is assigned} \\ 0 & \text{otherwise.} \end{cases}$$

The incorporation of absolute requirement compatibility (A_{km}) in the objective function expression will try to reduce the inter-cell movements. For this purpose, absolute requirement compatibility (ARC) should be calculated based on the number of operations.

The above model can also be written in the following equivalent form.

$$\text{Maximize} \sum_{k=1}^{p} \sum_{m=1}^{M} A_{km} x_{km}$$

Subject to

$$\sum_{k=1}^{p} x_{km} \leq |N_m| \qquad m = 1, \ldots, M$$

$$\sum_{m=1}^{M} x_{km} + u_k = M_U \qquad k = 1, \ldots, p$$

$$u_k + v_k = M_U - M_L \qquad k = 1, \ldots, p;$$

$$\sum_{k=1}^{p} v_k \leq M_T - pM_L$$

$$u_k,\ v_k \geq 0 \text{ and integer} \qquad k = 1, \ldots, p$$

$$x_{km} = 0 \text{ or } 1 \qquad k = 1, \ldots, p;\ m = 1, \ldots, M$$

The variables u_k and v_k are the surplus and slack variables associated with the constraints (3.116) and (3.117), respectively.

The formulation given above possesses totally unimodular matrix, and has the structure of a unbalanced transportation problem except for the constraint on the value of x_{km} which can only be binary. Because of this reason, it may not be simple to find the solution to the this problem.

3.7 NUMERICAL EXAMPLES

In this section, certain examples are presented to illustrate the various models developed in the Section 3.5. These problems are solved on IBM compatible PC/AT using LINDO software. The relative requirement compatibility appearing in the objective function expression of the various models is computed based on the number of the operations.

To view the effect of the constraints on the maximum and the minimum number of parts, machine capacity and cell disjointedness, let us consider an example (Example 3.4) for which the required data are given in Table 3.6(a) to 3.6(c). In the Table 3.6(c), other possible alternative process plans for the various parts are

Table 3.6(a): Value of Certain Parameters (Example 3.4).

Parameter	Notation	Value
Number of groups to be formed	p	2
Total number of process plans	q	15
Total number of machine types	M	5
Total number of parts (N)	N	7
Minimum number of machines to be assigned to various groups	M_L	3
Maximum number of machines which can be in the system	M_T	8
Maximum number of machines that can be assigned to a group	M_U	5
Minimum number of parts to be assigned to various groups	P_L	3
Maximum number of parts that can be assigned to a group	P_U	4

Table 3.6(b): Details of Machines (Example 3.4).

Machine type (m)	Number of machines ($\|N_m\|$)	Set of machines (N_m)	Capacity of each machine (T_m)	Total capacity (C_m)	Cost of a single machine (f_m)
1	2	{A1, A2}	3.00	6.00	10.0 lacs
2	2	{B1, B2}	3.00	6.00	11.0 lacs
3	2	{C1, C2}	3.00	6.00	12.0 lacs
4	2	{D1, D2}	3.00	6.00	13.0 lacs
5	1	{E1}	3.00	6.00	14.0 lacs

Table 3.6(c): Details of Process Plans for Various Parts (Example 3.4).

Part (n)	Process plan ($i \in F_n$)	(Machine type B(i,k), Processing time t(i,k)) required for operation k in process plan i: 1	2	3	4
1	1	(1, 1.50)	(2, 1.00)	(3, 1.00)	(4, 1.00)
	2	(1, 1.50)	(2, 1.00)	(3, 0.75)	(3, 0.75)
	3	(2, 1.20)	(2, 0.80)	(3, 0.75)	(3, 0.75)
2	4	(1, 1.00)	(2, 1.00)	(3, 1.00)	(4, 1.00)
	5	(1, 1.00)	(1, 1.00)	(3, 0.75)	(3, 0.75)
3	6	(3, 1.00)	(4, 1.00)	(5, 1.00)	
	7	(5, 0.50)	(4, 1.00)	(5, 1.00)	
4	8	(3, 1.00)	(4, 1.00)	(5, 1.00)	
	9	(5, 0.50)	(5, 0.50)	(5, 0.50)	
5	10	(1, 1.00)	(2, 1.00)	(3, 1.00)	
	11	(3, 0.50)	(3, 0.50)	(3, 1.00)	
6	12	(1, 1.00)	(3, 1.00)		
	13	(3, 0.75)	(3, 0.75)		
7	14	(2, 1.00)	(3, 1.00)		
	15	(4, 0.75)	(4, 0.75)		

not listed simply because of the brevity of the problem, and a better analysis and understanding of the solution. Different problem scenarios considered are given in Table 3.7. The grouping solution for these scenarios are presented in Table 3.8, and the other statistics in Table 3.9.

Table 3.7: The Details of the Problem Scenarios Considered for Example 3.4.

Problem scenario	Restriction on				
	Number of parts in a group		Machine capacity	Group disjointedness	Model to be used
	Maximum	Minimum			
1	-	-	-	-	M3.5(a)
2	-	Y	-	-	M3.5(b)
3	Y	-	-	-	M3.5(c)
4	Y	Y	-	-	M3.5(d)
5	-	-	Y	-	M3.5(e)
6	-	-	-	Y	M3.5(f)
7	-	-	Y	Y	M3.5(g)
8	Y	Y	Y	Y	M3.5(h)

Table 3.8: Grouping Solution for Different Problem Scenarios for Example 3.4.

Problem scenario	Process family	Part family	Cell of machine type	Objective function value
1	{2,5} {6,9,11,13,15}	{1,2} {3,4,5,6,7}	-	7.0
2	{2,4,12} {6,8,11,15}	{1,2,6} {3,4,5,7}	-	7.0
3	{1,4,11,15} {6,8,13}	{1,2,5,7} {3,4,6}	-	7.0
4	{3,5,10,13} {6,8,15}	{1,2,5,6} {3,4,7}	-	7.0
5	{1,5,10,12,15} {7,9}	{1,2,5,6,7} {3,4}	-	7.0
6	{2,5,10,13} {7,9,15}	{1,2,5,6} {3,4,7}	{1,2,3} {4,5}	7.0
7	{2,5,10,12,14} {7,9}	{1,2,5,6,7} {3,4}	{1,2,3} {4,5}	7.0
8	{3,5,11,12} {7,9,15}	{1,2,5,6} {3,4,7}	{1,2,3} {4,5}	7.0

Table 3.9: Load on Machines for the Grouping Shown in Table 3.8 (Example 3.4).

Problem scenario	Load on machine type				
	1	2	3	4	5
1	3.50	1.00	7.50	2.75	2.50
2	3.50	2.00	7.50	4.75	2.00
3	2.50	2.00	7.50	5.75	2.00
4	3.00	3.00	7.50	3.75	2.00
5	5.50	2.00	4.50	3.75	3.00
6	2.50	2.00	5.50	2.75	3.00
7	5.50	3.00	6.00	1.00	3.00
8	3.00	2.00	6.00	2.75	3.00

From the solution of the problem scenarios 1 to 4, the effect of the constraints on the maximum and the minimum number of parts can be seen. It should be noticed that the solutions obtained for the problem scenarios 2, 3 and 4 are the optimal solutions for any of these scenarios, e.g. the solution for the problem scenario 2 is also optimal for the scenarios 3 and 4.

Further, the load on the machines of type 3 for the scenarios 1 to 4 is 7.5 and is greater than the available capacity of 6.0. However, the introduction of capacity constraint brings the load on the machines within the available capacities as can be seen from the scenarios 5 to 8. The groups resulted in the scenarios 1 to 5 will not be disjoint if all the copies of a machine type are assigned to only one group. However, with the introduction of the constraint on group disjointedness, as in the scenarios 6 to 8, perfect grouping solution is obtained.

Now a different example (Example 3.5) is considered to illustrate the use of models M3.6(a), M3.6(d), M3.6(e), M3.8(a) and M3.8(b). The data for Example 3.5 are given in Table 3.10(a) to 3.10(c). In this case, all the possible alternative process plans for each of the parts are listed. The solutions obtained from the use of different model are presented in Table 3.11.

For the problems solved using the models M3.6(a) and M3.8(a), N_m given in the Table 3.10(b) is assumed to be the set of machines available of type m; while in use of the models M3.6(d) and M3.8(b), $|N_m|$ is assumed to represent the number of maximum machines of type m which can be procured.

Table 3.10(a): Values of Certain Parameters (Example 3.5).

Parameter	Notation	Value
Number of groups to be formed	p	2
Total number of process plans	q	10
Total number of machine types	M	4
Total number of parts (N)	N	5
Minimum number of machines to be assigned to various groups	M_L	1
Maximum number of machines which can be in the system	M_T	8
Maximum number of machines that can be assigned to a group	M_U	5
Minimum number of parts to be assigned to various groups	p_L	1
Maximum number of parts that can be assigned to a group	p_U	3

Table 3.10(b): Details of Machines (Example 3.5).

Machine type (m)	Number of machines ($\|N_m\|$)	Set of machines (N_m)	Capacity of each machine (T_m)	Total Capacity (C_m)	Cost of a single machine (f_m)
1	3	{A1, A2, A3}	3.00	9.00	10.0 lacs
2	3	{B1, B2, B3}	3.00	9.00	11.0 lacs
3	2	{C1, C2}	3.00	6.00	12.0 lacs
4	2	{D1, D2}	3.00	6.00	13.0 lacs

Table 3.10(c): Details of Process Plans for Various Parts (Example 3.5).

Part (n)	Process plan ($i \in F_n$)	(Machine type B(i,k), Processing time t(i,k)) required for operation k in process plan i: 1	2
1	1	(1, 2.00)	(2, 2.00)
	2	(2, 2.00)	(2, 2.00)
2	3	(1, 2.00)	(3, 2.00)
	4	(1, 2.00)	(2, 2.00)
3	5	(1, 2.00)	(4, 2.00)
	6	(1, 2.00)	(2, 2.00)
4	7	(3, 2.00)	(4, 2.00)
	8	(2, 2.00)	(4, 2.00)
5	9	(3, 2.00)	(4, 2.00)
	10	(1, 2.00)	(4, 2.00)

Table 3.11: Grouping Solution for Example 3.5 Considering Process plans.

Model Used	Modified Value of Parameters	Process Family		Part Family		Machine Cell m/c type(detail)		Objective Function Value Considering	
		1	2	1	2	1	2	RRC	Investment
M6(a)	None	1,4,6	7,9	1,2,3	4,5	1(2 No.) 2(2 No.)	3(2 No.) 4(2 No.)	5.0	-
M6(d)	None	1,4,6	7,9	1,2,3	4,5	1(2 No.) 2(2 No.)	3(2 No.) 4(2 No.)	-	92.00 lacs
M6(e)	$N_m = 1 \ \forall m$	1,4,6	7,10	1,2,3	4,5	1(2 No.) 2(2 No.)	1(1 No.) 3(1 No.) 4(1 No.)	-	49.00 lacs
M8(a)	None	1,4,6	7,9	1,2,3	4,5	1(A1,A2) 2(B1,B3)	3(C1,C2) 4(D1,D2)	5.0	-
M8(b)	None	1,4,6	7,9	1,2,3	4,5	1(A1,A2) 2(B1,B2)	3(C1,C2) 4(D1,D2)	-	92.00 lacs

For the problem solved using model M3.6(e), $|N_m|$ is assumed to represent the number of available machines of type m and is equal to one for all the machine types. The difference between the number of machines assigned to various cells and that already available, gives the number of additional machines of various types which are to be procured. The additional machines of type 1, 2 and 4 are 2, 1 and 4, respectively.

Example 3.5 is also used to illustrate models M3.7(a) and M3.7(b). The use of these models requires input about route plans. The route plans corresponding to Example 3.5 are given in Table 3.12. The M_T is taken to be equal to 10; or in other words, the constraint on total number of machines was made ineffective. The solution to this problem is presented in Table 3.13.

It can be seen from the input data and the machine capacity constraint (72) that the route plans 10, 14 and 18 will never be selected for assignment. Further, on comparing the solutions shown in the Tables 3.11 and 3.13 and obtained using the respective models M3.8(b) and M3.7(b), it can be observed that the model M3.7(b) does not make use of the remaining capacity of the machines of type 1 and 2 and asks for extra investment of Rs. 21.00 lacs. Besides the bad solution, the model M3.7(b) requires 68 route plans to be considered as compared to only 10 process plans considered by the model M3.8(b), and handles unnecessarily a much larger size problem. This illustration clearly establishes inferiority of the model M3.7(b) over M3.8(b).

Table 3.12: Details of Route Plans for Various Parts (Example 3.5).

Part	Route Plan	Operation 1	Operation 2
1	1	(A1,2.0)	(B1,2.0)
	2	(A1,2.0)	(B2,2.0)
	3	(A1,2.0)	(B3,2.0)
	4	(A2,2.0)	(B1,2.0)
	5	(A2,2.0)	(B2,2.0)
	6	(A2,2.0)	(B3,2.0)
	7	(A3,2.0)	(B1,2.0)
	8	(A3,2.0)	(B2,2.0)
	9	(A3,2.0)	(B3,2.0)
	10	(B1,2.0)	(B1,2.0)
	11	(B1,2.0)	(B2,2.0)
	12	(B1,2.0)	(B3,2.0)
	13	(B2,2.0)	(B1,2.0)
	14	(B2,2.0)	(B2,2.0)
	15	(B2,2.0)	(B3,2.0)
	16	(B3,2.0)	(B1,2.0)
	17	(B3,2.0)	(B2,2.0)
	18	(B3,2.0)	(B3,2.0)
2	19	(A1,2.0)	(C1,2.0)
	20	(A1,2.0)	(C2,2.0)
	21	(A2,2.0)	(C1,2.0)
	22	(A2,2.0)	(C2,2.0)
	23	(A3,2.0)	(C1,2.0)
	24	(A3,2.0)	(C2,2.0)
	25	(A1,2.0)	(B1,2.0)
	26	(A1,2.0)	(B2,2.0)
	27	(A1,2.0)	(B3,2.0)
	28	(A2,2.0)	(B1,2.0)
	29	(A2,2.0)	(B2,2.0)
	30	(A2,2.0)	(B3,2.0)
	31	(A3,2.0)	(B1,2.0)
	32	(A3,2.0)	(B2,2.0)
	33	(A3,2.0)	(B3,2.0)
3	34	(A1,2.0)	(D1,2.0)
	35	(A1,2.0)	(D2,2.0)
	36	(A2,2.0)	(D1,2.0)
	37	(A2,2.0)	(D2,2.0)
	38	(A3,2.0)	(D1,2.0)
	39	(A3,2.0)	(D2,2.0)
	40	(A1,2.0)	(B1,2.0)
	41	(A1,2.0)	(B2,2.0)
	42	(A1,2.0)	(B3,2.0)
	43	(A2,2.0)	(B1,2.0)
	44	(A2,2.0)	(B2,2.0)
	45	(A2,2.0)	(B3,2.0)
	46	(A3,2.0)	(B1,2.0)
	47	(A3,2.0)	(B2,2.0)
	48	(A3,2.0)	(B3,2.0)
4	49	(C1,2.0)	(D1,2.0)
	50	(C1,2.0)	(D2,2.0)
	51	(C2,2.0)	(D1,2.0)
	52	(C2,2.0)	(D2,2.0)
	53	(B1,2.0)	(D1,2.0)
	54	(B1,2.0)	(D2,2.0)
	55	(B2,2.0)	(D1,2.0)
	56	(B2,2.0)	(D2,2.0)
	57	(B3,2.0)	(D1,2.0)
	58	(B3,2.0)	(D2,2.0)
5	59	(C1,2.0)	(D1,2.0)
	60	(C1,2.0)	(D2,2.0)
	61	(C2,2.0)	(D1,2.0)
	62	(C2,2.0)	(D2,2.0)
	63	(A1,2.0)	(D1,2.0)
	64	(A1,2.0)	(D2,2.0)
	65	(A2,2.0)	(D1,2.0)
	66	(A2,2.0)	(D2,2.0)
	67	(A3,2.0)	(D1,2.0)
	68	(A3,2.0)	(D2,2.0)

Table 3.13: Solution to Example 3.5 Considering Route Plans.

Model used	Process family	Part family	Machine cell	Objective function value
M7(a)	{1,29,48} {49,62}	{1,2,3} {4,5}	{A1,A2,A3,B1,B2,B3} {C1,C2,D1,D2}	7.00*
M7(b)	{1,29,48} {49,62}	{1,2,3} {4,5}	{A1,A2,A3,B1,B2,B3} {C1,C2,D1,D2}	113.00 Lacs**

: Value for the objective of maximization of the sum of RRC.
: Value for the objective of minimization of the total investment

3.8 SUMMARY AND CONCLUSIONS

In this chapter, the formulations of the grouping problem proposed by Kusiak (1987) and Shtub (1989) are modified resulting comparatively lesser number of variables and constraints. It is shown by an example that the models of Kusiak and Shtub may not necessarily result perfect grpuping. The grouping problem is extended to consider identical machines with or without aggregation allowing two or more operations in any process plan to use the same type of machine. It is shown that the consideration of capacities of machines do play an important role in case of generalized grouping with parts having alternative process plans. The grouping models developed incorporating the constraints on capacities of the machines not only provide groups of parts and machines, but also solutions to the loading problems.

The models have also been developed to integrate the generalized grouping problem with the production system design problem for determining the number of different kind of machines assigned to different cells of production system for cellular manufacturing.

The models developed in this chapter are NP-complete. Efforts have been made to develop heuristic approaches for solving such problems. Few of these are based on Lagrangian relaxation. Lagrangian relaxation method has not been found suitable for the grouping problems at design level. To overcome the complexity for solving the problem, a hierarchical method for group formation is proposed.

CHAPTER IV

Generalized Grouping Problem :Basic Problem As A p-Center Problem

4.1 INTRODUCTION

The grouping problem discussed in Section 3.2 of the chapter III has the objective of maximizing the sum of similarity of process plans computed with respect to the process plan representing the corresponding process family. For such a problem, the optimal solution may contain a process plan whose similarity with the plan representing the corresponding process family may be quite low. However, a resonable grouping proceedure should not allow the possibility of such occurrences. In such situations, it may be desirable to maximize the minimum similarity of a plan with respect to the plan representing the process family. In case where all the parts have single process plan, the problem will be equivalent to a p-center problem. In a p-center problem, the objective is to choose as center p nodes from the total N nodes of a graph and associate every node out of (N-p) nodes with one and only one center such that the minimum advantage of such association gets maximized (Minieka (1977)). For solving these basic p-center problems, a number of approaches have been suggested by several researchers (Chen (1983), Christofides and Viola (1977), Minieka (1970)). In the context of present grouping problem, p-centers to be found are equivalent to p parts representing p part families, and N nodes are equivalent to total N parts considered for grouping. Thus the equivalent p-center problem

seeks to maximize the minimum similarty of a part with the part representing the corresponding family.

In the general case where a part has several alternative process plans and out of which only one process plan is to be chosen in process family formulation, the grouping problem will be an extended form of basic p-center problem. There dose not seem to exist any methodology which can directly be used for solving such a generalized version of p-center problem. For this purpose, a heuristic approach based on the concepts used in the heuristic described in the Section 6.1.1 of the Chapter III has been proposed.

In the following sections, first the formulation for the generalized grouping problem is presented, and then heuristic approach to be used for solving this problem is described. Thereafter, examples are given to illustrate the various steps of the heuristic. In the last section, summary of this approach is presented.

4.2 PROBLEM STATEMENT

The problem considered is the formation of prespecified number of groups of process plans while maximizing the minimum relative requirement capability of a plan computed with respect to the plan representing the corresponding process family, and selecting only one process plan for part from its several alternatives.

For the mathematical representation of the above problem, the notations introduced in the Sections 3.4.1 and 3.6.1 of the Chapter III are used with the same meaning and definitions.

The problem can be expressed as:

MODEL M4.1

$$\text{Maximize} \left[\underset{n}{\text{minimum}} \left\{ \underset{i \in F_n;\ j}{\text{maximum}} \ (R_{ij}\ x_{ij}) \right\} \right]$$

Subject to:

$$\sum_{i \in F_n} \sum_{j=1}^{q} x_{ij} = 1 \qquad n = 1, \ldots, N$$

$$\sum_{j=1}^{q} x_{jj} = p$$

$$\sum_{i=1}^{q} x_{ij} \leq (N-p+1)\ x_{jj} \qquad j = 1, \ldots, q$$

$$x_{ij} = 0 \text{ or } 1 \qquad i,j = 1, \ldots, q.$$

The above constraints are ths same as for the model M3.3 of the Chapter III.

4.3 SOLUTION METHODOLOGY

It can be seen that the problem described by the model M4.1 is hard to be solved using any mathematical programming methodology. For this problem, a heuristic approach is suggested which is similar to that given in the Section 3.6.1.1. Various steps of the heuristic are given in Section 4.3.1, while the method for solving the mathematical models appearing in the heuristic in Section 4.3.2. Some comments on the relevance of various steps of this method are presented in Section 4.3.3, and on its computational complexity in Section 4.3.4.

4.3.1 Heuristic for Maximizing the Minimum Relative Requirement Compatibility

Various steps of the heuristic are given below. A flowchart

showing the detail of the heuristic is given in Figure 4.1.

<u>Selection of p processplans for representing process families</u>

Step 1: For each process plan i, find maximum RRC of process plans of each of the other parts computed with respect to plan i. Then for this plan i, find the total number of parts for which the maximum of RRC with respect to the plan i is positive, and denote this number by S_i. Thus,

$$S_i = \sum_{\substack{n=1 \\ n \neq \underline{n}}}^{N} \mu \left\{ \underset{j \in F_n}{\text{maximum}} \left(R_{ji} \mid R_{ji} > 0.0 \right) \right\} \qquad \forall\ i \in F_{\underline{n}} \text{ and } \underline{n} = 1, \ldots, N.$$

Step 2: Sequence the process plans in decreasing order of the value of S_i. In case of a tie, plans should be arranged in the order of decreasing value of the lowest positive RRC of process plans of the other parts. This number for a process plan $i \in F_{\underline{n}}$ is equal to

$$\underset{\substack{n \\ n \neq \underline{n}}}{\text{minimum}} \left\{ \underset{j \in F_n}{\text{minimum}} \left(R_{ji} \mid R_{ji} > 0.0 \right) \right\}.$$

Further tie should be broken by sequencing the plans in the order of decreasing number of the total machine types required. In case of further tie, plans should be arranged in the order of decreasing value of the largest positive RRC of process plans of the other parts. This number for a process plan $i \in F_{\underline{n}}$ is equal to

$$\underset{\substack{n \\ n \neq \underline{n}}}{\text{minimum}} \left\{ \underset{j \in F_n}{\text{maximum}} \left(R_{ji} \mid R_{ji} > 0.0 \right) \right\}.$$

Step 3: The first p process plans in the sequence and each belonging to a different part will represent p process

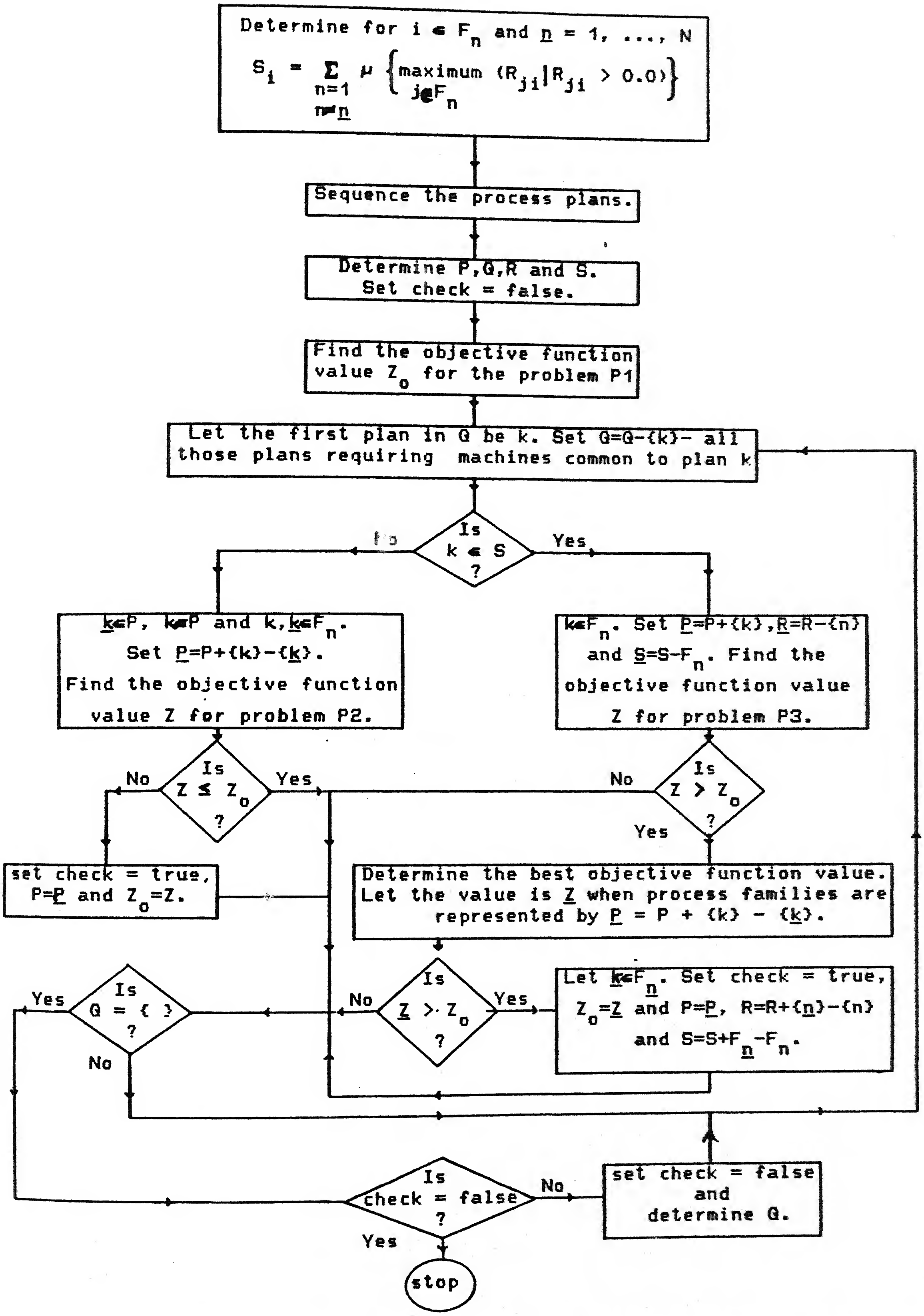


Figure 4.1 : Flow Chart for the Heuristic for Determining the Groups While Maximizing the Minimum of the Similarity Value.

families. A process plan representing a process family should not have all its machine requirements common to some other process plans chosen for representing process families. This should be allowed only when no plan is available for such consideration. Now let,

P = set of process plans representing process families

Q = set of remaining process plans (i.e. $\notin$ P) whose machine requirements are not common to the plans in P

R = set of those parts of which no process plan represents any process family

S = set of process plans of parts in R

The plans in Q are arranged according to the sequence determined in Step 2.

Set check = false and go to the next step.

Determination of initial grouping solution

Step 4: Solve the problem P4.1, given below.

Problem P4.1

$$\text{Maximize}\left[\min_{n\in R}\left\{\max_{i\in F_n;\, j\in P} (R_{ij}\, x_{ij})\right\}\right]$$

Subject to:

$$\sum_{i\in F_n}\sum_{j\in P} x_{ij} = 1 \qquad \forall\, n \in R$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall\, n \in R;\ \forall\, i \in F_n \text{ and } \forall\, j \in P$$

Let the objective function value be Z_o. Go to Step 10.

Perturbation for improving the current solution

Step 5: Let the first plan in Q be k. Set Q = Q − {k}. Remove from Q those plans also whose machine requirements are

completely common to that of plan k. If $k \in S$, go to Step 7 else to the next step.

Step 6: Let k be the process plan of a part of which some other process plan $\underline{k}$ is in P. That is, $\underline{k} \in P$, $k \notin P$, and k, $\underline{k} \in F_n$.

Set $\underline{P} = P + \{k\} - \{\underline{k}\}$. Solve the problem P4.2 given below.

Problem P4.2

$$\text{Maximize}\left[\underset{n \in R}{\text{minimum}}\left\{\underset{i \in F_n; j \in \underline{P}}{\text{maximum}} (R_{ij}\ x_{ij})\right\}\right]$$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in \underline{P}} x_{ij} = 1 \qquad \forall\ n \in R$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall\ n \in R;\ \forall\ i \in F_n \text{ and } \forall\ j \in \underline{P}$$

Let the objective function value for the above problem be Z. If $Z \leq Z_o$, then go to Step 10. Otherwise, set check = true, $P = \underline{P}$, $Z_o = Z$ and go to Step 10.

Step 7: Let the process plan k belongs to part n, i.e. $k \in F_n$. Set $\underline{P} = P + \{k\}$, $\underline{R} = R - \{n\}$ and $\underline{S} = S - F_n$. Solve the problem P4.3 given below.

Problem P4.3

$$\text{Maximize}\left[\underset{n \in \underline{R}}{\text{minimum}}\left\{\underset{i \in F_n; j \in \underline{P}}{\text{maximum}} (R_{ij}\ x_{ij})\right\}\right]$$

Subject to:

$$\sum_{i \in F_n} \sum_{j \in \underline{P}} x_{ij} = 1 \qquad \forall\ n \in \underline{R}$$

$x_{ij} = 0$ or 1 $\quad\quad \forall\ n \in \underline{R};\ \forall\ i \in F_n$ and $\forall\ j \in \underline{P}$

Let the objective function value for the problem P4.3be Z. If $Z > Z_o$, then go to the next step else to Step 10.

Step 8: Find the groups of process plans where one of the process family is represented by process plan k and rest (p-1) families by (p-1) process plans belonging to P. Let the maximum value of the objective function of these problems be $\underline{Z}$ for $\underline{P} = P + \{k\} - \{\underline{k}\}$. If $\underline{Z} > Z_o$, then go to the next step else go to Step 10.

Step 9: Set check = true, $Z_o = \underline{Z}$ and $P = \underline{P}$. Let $\underline{k} \in F_{\underline{n}}$, then set $R = R + \{\underline{n}\} - \{n\}$ and $S = S + F_{\underline{n}} - F_n$ and go to the next step.

Termination of procedure

Step 10: If Q is not empty, go to Step 5. Otherwise, go to the next step.

Step 11: If check = false, then stop. Z_o will be the optimal objective function value.

Otherwise, set check = false, Q as the set of all process plans not in P, and go to Step 5.

4.3.2 Method for Solving Mathematical Models in the Heuristic

The problems described by P4.1, P4.2, P4.3 and also those at the Step 8 are having the same structure, and thus solution methodology for each of them will remain the same. In the following paragraph, method for solving P4.1 is given which after suitable modifications can be used for solving the problems P4.2,

P4.3 and the other problems encountered at the Step 8 of the heuristic.

The objective of maximizing the following expression

$$\left[\underset{n \in R}{\text{minimum}} \left\{ \underset{i \in F_n; j \in P}{\text{maximum}} (R_{ij}\ x_{ij}) \right\} \right]$$

subject to the constraint $\sum_{i \in F_n} \sum_{j \in P} x_{ij} = 1 \quad \forall\ n \in R$ is the same as finding for each part belonging to R, a process plan and a group for which the relative requirement compatibility of this process plan computed with respect to a process plan representing the corresponding group is the maximum. It is equivalent to:

(i) finding $R_{\underline{i},\underline{j}} = \underset{i \in F_n; j \in P}{\text{maximum}} (R_{ij})\ \forall\ n \in R$ and then

(ii) setting $x_{ij} = 0 \quad \forall\ i \in F_n$ and $\forall\ j \in P$ except for $x_{\underline{i},\underline{j}}$ which is set equal to one.

The objective function value will be equal to minimum of all such $R_{\underline{i},\underline{j}}$ computed.

4.3.3 Comments on Various Steps of the Heuristic

Most of the steps of this heuristic are similar to that described in the Section 3.6.1.1, and the basic idea behind their use is also the same. The steps 1, 2 and 3 try to select in somewhat greedy manner a set of p process plans to represent p process families. In the Step 2, process plans are sequenced in a particular way with the aim to result possibly the best solution. Arranging the plans in the decreasing order of the number of parts whose maximum RRC values with respect to them is positive, has a better chance of resulting higher objective function value. The tie resolved on the basis of the decreasing value of the lowest positive RRC of other parts indirectly ensures that RRC of the

other parts with respect to plans representing families would be high. Further tie broken based on the total number of different types of machine required, will naturally provide a better compatibility to the requirements of the other parts, and thus a higher objective function value. In a good grouping, member process plans should have highest compatibility with the plan representing processs families. It is for this reason, subsequent ties are resolved based on the decreasing value of the largest positive RRC of process plans of the other parts.

4.3.4 Computational Complexity

The heuristic described in the Section 4.2.1 is similar to the heuristic described in the Section 3.6.1.1. The various steps of the two heuristics have almost the same computational requirements. Using the analysis similar to that given in the Section 3.6.1.4, it can be seen that the computational requirement of the present heuristic is also polynomially bounded.

4.4 EXAMPLES

The two examples given in the Section 3.6.1.5 of the Chapter III are taken for illustrating the use of the heuristic.

<u>Example 4.1</u>

This example is the same as the example 3.2. The steps of the heuristic followed for solving this example are as given below.

Step 1: For each plan the number of parts, for which the maximum of relative requirement compatibility of their alternative process plans are positive, is given by the following vector.

$\{S_i\} = \{6, 6, 6, 6, 6, 6, 4, 6, 1, 6, 6, 6, 6, 6, 4\}$.

Step 2: The sequence of the process plan is:

$\{1, 4, 6, 8, 2, 10, 3, 5, 12, 14, 11, 13, 7, 15, 9\}$.

Step 3: $P = \{1, 6\}$, $Q = \phi$, $R = \{2, 4, 5, 6, 7\}$,

$S = \{4, 5, 8, 9, 10, 11, 12, 13, 14, 15\}$ and

check = false.

Step 4: Process families are $\{1, 4, 10, 12, 14\}$ and $\{6, 8\}$.

$Z_o = 1.00$. Now the Step 10 is to be followed.

Step 10: Since Q is empty, the Step 11 is to be followed.

Step 11: Since check = false, the procedure is terminated.

Present process families $\{1, 4, 10, 12, 14\}$ and $\{6, 8\}$ with the objective function value equal to one will represent the optimal solution.

Example 4.2

The second example considered is the example 3.3. Following are the steps followed of the heuristic stated in the Section 4.2.1 to solve this example.

Step 1: Following vector represents for each plan, the number of parts for which the maximum of relative requirement compatibility of their alternative process plan is positive:

$\{S_i\} = \{4, 3, 4, 4, 4, 4, 3, 4, 3, 4\}$.

Step 2: The sequence of the process plan is:

$\{1, 4, 6, 8, 10, 3, 7, 9, 2\}$.

Step 3: $P = \{1, 5\}$, $Q = \{8, 3, 7, 9\}$, $R = \{2, 4, 5\}$,

$S = \{3, 4, 7, 8, 9, 10\}$ and check = false.

Step 4: Process families obtained are $\{1, 4\}$ and $\{5, 7, 10\}$, with the objective function value $Z_o = 0.5$. The Step 10 is to

be followed.

Step 10: Since Q is not empty, the Step 5 will be followed.

Step 5: $k = 8$ and $Q = \{3, 7, 9\}$. Since $k \in S$, the Step 7 is to be followed.

Step 7: $\underline{P} = \{1, 5, 8\}$. Since $k \in F_4$ (i.e. $n = 4$), $\underline{R} = \{2, 5\}$ and $\underline{S} = \{3, 4, 9, 10\}$. Process families determined are {1,4}, {5, 10} and {8} with the objective function value Z equal to 1.00. Since $Z > Z_o$, the Step 8 will be followed.

Step 8: The two sets of process families and the corresponding objective function values are as follows.

Set 1: Process families are {1, 4, 6} and {8, 9} with $Z = 0.50$.

Set 2: Process families are {5, 3, 10} and {8, 2} with $Z = 0.50$.

Since the objective function value for each of the two sets of groups are the same as Z_o, the Step 10 is to be followed.

Step 10: Since Q is not empty, the Step 5 is to be followed.

Step 5: $k = 3$ and $Q = \{7, 9\}$. Since $k \in S$, the Step 7 is to be followed.

Step 7: Again no improvement is observed. Finally, the Step 5 has to be followed.

Step 5: $k = 7$ and $Q = \phi$. Since $k \in S$, the Step 7 is to be followed.

Step 7: $\underline{P} = \{1, 5, 7\}$. Since $k \in F_4$ (i.e. $n = 4$), $\underline{R} = \{2,5\}$ and $\underline{S} = \{3, 4, 9, 10\}$.

Process families obtained are {1, 2}, {5} and {7, 9} with the objective function value Z equal to 1.00. Since $Z >$

Z_o, the Step 8 is to be followed.

Step 8: The two sets of process families and the corresponding objective function values are as follows.

Set 1: Process families are {1, 4, 6} and {7, 9} with Z=1.00.

Set 2: Process families are {5, 1, 3} and {7, 9} with Z=0.50.

Therefore, $\underline{Z}$ = 1.00 for $\underline{P}$ = {1, 7}. Since $\underline{Z} > Z_o$, the Step 9 is to be followed.

Step 8: Set check = true, Z_o = 1.00 and P = {1, 7}. Since $\underline{k}$ = 5 and $\underline{k} \in F_3$, thus R = {2, 3, 5} and S = {3,4,5,6,9,10}. Now the Step 10 will be followed.

Step 10: Since Q is empty, the Step 11 is to be followed.

Step 11: Since check is not equal to false, the Step 5 onwards will be followed with Q = {3, 5, 8, 10} and check = false.

On following the further steps of the heuristic, no improvement is observed into the objective function value (which is already at its maximum of 1.00). Present groups {1, 4, 6} and {7, 9} with the objective function value equal to one remain optimal.

It is seen that the solutions for the two examples 3.2 and 3.3 obtained using the heuristics described in the Sections 3.6.1.1 and 4.2.1 are the same. However, the two heuristics, in general, may not necessarily yield the same solution. For example, consider the problem with details of process plans and relative requirement compatibility given in Figure 4.2.

		Part 1	Part 2		Part 3
	Process Plan	1	2	3	4
Part 1	1	1.00	0.50	0.44	0.70
Part 2	2	0.64	1.00	0.50	0.50
	3	0.57	0.50	1.00	0.70
Part 3	4	0.90	0.50	0.70	1.00

Figure 4.2: Relative Requirement Compatibility Matrix $[R_{ij}]$.

The Figure 4.2 contains only the partial information of the problem because it is assumed that the parts 1, 2 and 3 are closer to each other as compared to the other parts which together form (p-1) groups. The plans in each of these (p-1) groups are also assumed to be quite close to each other. The parts 1, 2 and 3, then, will form p^{th} group.

For the objective of maximization of the total relative requirement compatibility, the group will consist the process plans 1, 2 and 4, whereas for the objective of maximization of minimum relative requirement compatibility, it will consist plans 1, 3 and 4. The other details of the solutions are given in Table 4.1.

Table 4.1: Solution for the Example Shown in the Figure 4.2.

Process family	Index of plan representing process family	Sum of RRCs	Minimum RRC value
{1,2,4}	1	2.54	0.64
{1,3,4}	4	2.40	0.70

From the solution and the information given in the Table 4.1, it is obvious that the two heuristics will not necessarily yield the same solution. In case of process family {1, 2, 4}, RRC_{41} (= 0.90) though is high, RRC_{21}(= 0.64) is somewhat low. However, in case of process family {1, 3, 4}, both RRC_{14} (= 0.70) and RRC_{34} (= 0.70) are reasonably high.

4.4 SUMMARY AND CONCLUSIONS

In the present chapter, the generalized grouping problem is formulated as an extended version of p-centre problem where the minimum relative requirement compatibility of member process plans with respect to the plans representing corresponding process families, is maximized. Such consideration is shown to select those member plans whose requirements are close to plan representing the corresponding process family. For solving this problem, a heuristic approach is suggested.

The heuristic can suitably be modified to accommodate the constraints on the minimum and the maximum number of the parts that can be assigned to a group.

The hierarchical approaches used for group formation generally require use of some kind of similarity coefficient. The p-median formulation proposed by Kusiak (1987) and discussed in the Chapter III is an example in this regard. The p-median formulation in simple grouping finds groups of parts first; and in the generalized case, the groups of process plans. It has been observed in the chapter III through an illustrative example that the approach may not yield disjoint groups even though they exist. Since the main objective behind the grouping is to reduce the intercell movements and traffic congestion, it is desired that the groups be independent and disjoint as much as possible. For this purpose, a constraint on the group disjointedness was explicitly included into the basic p-median formulation. It has been observed that without this constraint, one can not be sure of obtaining disjoint groups. It has been further noticed that the use of similarity coefficients does not have much impact on disjoint group formation particularly in case of generalized grouping. This raises the suspicion regarding the effectiveness and importance of similarity coefficients for determination of part families and machine cells in generalized case irrespective of the approach used for the group formation.

In the present chapter, a different methodology has been adopted where instead of using similarity coefficients for group formation, a more natural method is followed. As mentioned earlier, a good grouping will require least intercell movements and allow maximum number of operations of parts to be completed by the machines belonging to the same group, and a good methodology should bring majority of the numbers representing association

between parts and machines within the blocks that represent various groups. It is this basic philosophy which has been used for group formation in the present chapter. The numbers representing association between parts and machines are binary (0 or 1) if the requirements of the parts on the various machines are not differentiated; and non-binary if done. These numbers are generally given in the form of a part-machine incidence matrix.

There are number of approaches reported in the literature that use the above mentioned philosophy. Production flow analysis of Burbidge (1971), rank order clustering of King and Nakornchai (1982), ideal seed clustering and modified rank order clustering (MODROC) of Chandrashekheran and Rajagopalan (1986, 1987, 1986b), occupancy-value method of Khator and Irani (1987), within-cell utilization based heuristic of Ballakur and Studel (1987), and assignment model of Srinivasan et al (1989) are examples in this regard. All of these approaches are heuristic in nature. The philosophy behind the mathematical models of grouping problems proposed by Kumar et al (1986) and Ventura et al (1987) is also similar. These models explicitly aim at collecting most of the numbers within the blocks.

Disjoint groups, as mentioned earlier, are ideal and are generally preferred. However, there may be certain situations in which complete independence of groups may not be possible and bringing disjointedness using additional resources may not be effective and economical. Moreover, in certain other cases, though procurement of binding resources may be advantageous, but because of certain other constraints such as budget, space available for accommodating machines, layout of materials handling

system, and also because of the characteristic of supervisory computers and control systems, etc., it may not be possible to do so. In situations like this, bottleneck machines which are required by the parts belonging to different groups need to be assigned properly. A bottleneck machine should be assigned to a group that brings maximum advantage. For making decision regarding assignment of machines under these circumstances, some relative weight of the part, production volume and related materials handling cost can be considered. Besides these part related characteristics, certain other factors which suitably weigh the association of parts with machines can also be considered. The association can be determined by identifying the tooling and operation compatibility of parts with machines (refer to Chapter III), processing costs and times of performing operations of parts on machines, nature of machines (required by most of the parts versus poorly required) and similarly several other factors.

Most of the approaches mentioned in the previous paragraphs use binary numbers. Ballakur and Studel (1987) and Ventura et al (1987) instead consider processing times of the operations of parts on various machines. Kumar et al (1986) include volume of parts to be processed on machines, or profit or productivity potential of parts associated with machines. Mosier and Taube (1985) associate weight with each of the parts showing their relative importance. Seiffodini (1988) incorporates production volumes of the parts and also the number of times a machine is required by a part.

These considerations though suggested basically for simple

grouping situation, can as well be extended and used for the generalized grouping situations. For a generalized case where an operation can be carried out on more than one machine, Gunasingh and Lashkari (1987a and 1989b) suggest measures based on tooling and operations compatibility between parts and machines. These measures themselves show some kind of association between parts and machines.

In the present work, somewhat more generalized framework has been considered where the association between a part and a machine is assumed to depend upon the operation involved. In other words, for an operation of a part the strength of the association is assumed to vary with the machines on which this operation can be performed. The association is assumed to include all the important and relevant factors mentioned in the previous paragraphs such as relative importance of parts and their production volumes, processing costs and times required for performing an operation on various alternative machines, tooling and setup requirements, ease of performing the operations, required versus attainable accuracy, etc.. The methodology adopted is centered at maximizing the association of the operations of the parts with respect to the machines belonging to the same group and also to which the operations are finally assigned. The problem is shown to be similar to that of minimizing the intercell movement cost when these numbers reflect material handling cost each corresponding to a combination of an operation of a part and the machine on which this operation can be performed.

In the grouping problems discussed in this chapter decisions regarding assignment of operations are also taken while deciding

the groups. The operations assignment problem was indirectly solved in case of generalized grouping modeled in the previous chapter where the process plans (route plans) selected in the group formation provided information regarding the assignment of operations on machines. On the contrary, in the grouping methodologies to be discussed later in this chapter, the solution to the operations assignment problem is found directly.

The grouping problems envisaged in this chapter include several practical constraints which have been discussed in detail in the previous chapter. To mention a few, these are: the constraints on the maximum and the minimum number of parts and machines to be assigned to each group, the number of machines of each type and their capacity. The constraint on the total number of machines in the system has also been considered but for the grouping at the planning level. The grouping objective, in this situation, is to minimize the total investment on the machines in FMS.

It should be noted that Kumar et al (1986) and also Ventura et al (1987) do not differentiate between part and machine while considering the constraint on size of the group, and the size of a group is defined as the total number of parts and machines assigned to it. Such a consideration though may satisfy the group size constraint but may lead to a situation where a group can contain few parts and large number of machines and vice versa, or only parts or machines. Having groups of only parts or machines does not make much sense and similarly the unbalanced composition of parts and machines in a group. Thus, formation of such groups should be avoided. In the present work, this discrepancy does not

Decision Variables

$$x_{nimk} = \begin{cases} 1 & \text{if part } n \text{ and machine type } m \text{ are assigned to group } k \text{ and the operation } i \text{ of part } n \text{ is assigned to machine type } m \\ 0 & \text{otherwise} \end{cases}$$

$$y_{nk} = \begin{cases} 1 & \text{if part } n \text{ is assigned to group } k \\ 0 & \text{otherwise} \end{cases}$$

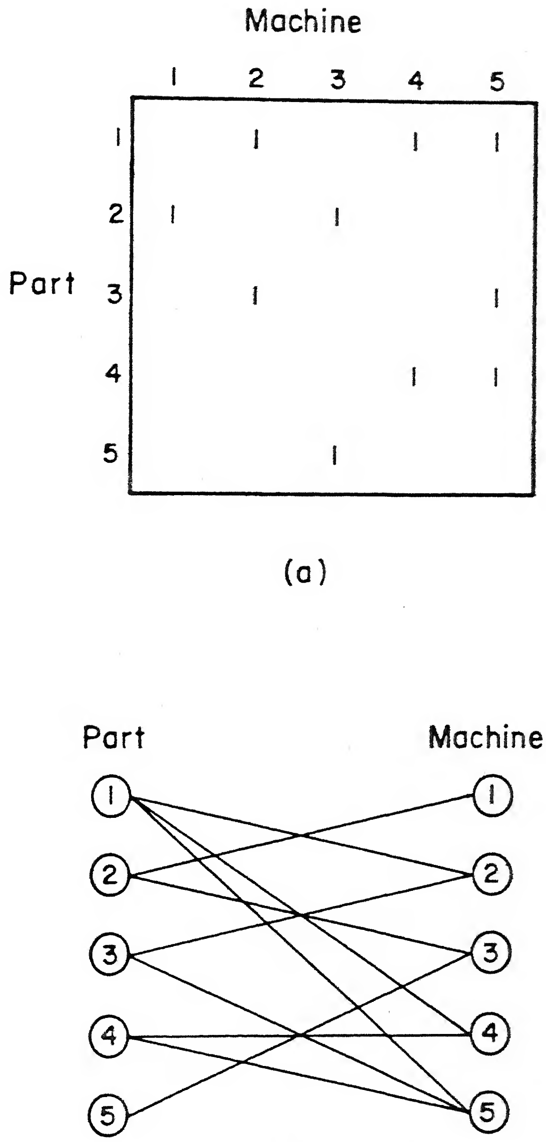
Z_{mk} = number of machines of type m assigned to group k

All the operations are numbered differently, i. e. no two operations whether on the same part or on different can have the same index.

5.3 GROUPING APPROACH

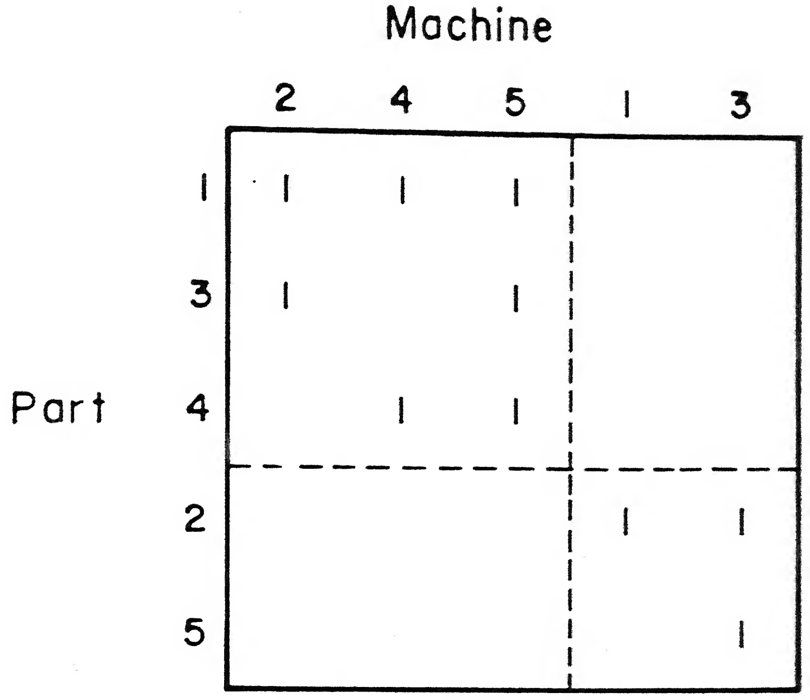
It can be recalled that the mathematical models developed in the previous chapter are typically extensions of a basic model that represents formulation for a p-median problem. However, in the present chapter, the basic problem resembles closely with the problem of bipartite graph partitioning as can be seen from the following.

Consider a problem (Example 5.1) of simple grouping corresponding to which the part-machine matrix and equivalent bipartite graph representation is given in Figure 5.1. The matrix and the corresponding graph after rearrangement are shown in Figure 5.2. From these figures, it can be seen that the problem of finding given number of groups is equivalent to decomposing the corresponding bipartite graph in the same number.

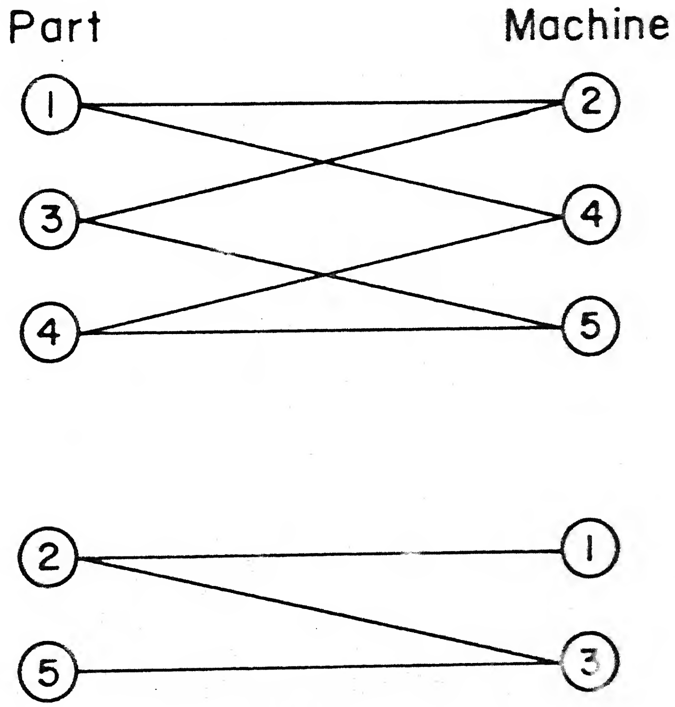


(b)

Figure 5.1: Initial (a) Part-Machine Matrix
(b) Corresponding Bipartite Graph (Example 5.1).



(a)



(b)

Figure 5.2: Rearranged (a) Part-Machine Matrix
(b) Bipartite Graph (Example 5.1).

In the generalized grouping situation also, the basic philosophy for group determination remains the same. The graph, of course, is not bipartite and can best be represented by an AND-OR graph. For illustration, consider the example (Example 5.2) represented by AND-OR graph shown in Figure 5.3. There are four parts and five machines. The number of operations to be performed on the parts are 4, 4, 3 and 3, respectively. There are certain operations which can be performed on only one machine, whereas some other on more than one. For example, operation 1 can be performed on either of the machines 1 and 2, whereas operation 2 only on the machine 2. Thus, the edges emanating from a node at Level 1 (see the Figure 5.3) will represent AND condition, and that from a node at Level 2 will represent OR condition.

For the case when two groups of parts and machines are to be determined each being nonempty, a possible solution will be:

Part family 1 = {1,2},

Part family 2 = {3,4},

Machine cell 1 = {1,2,3},

and Machine cell 2 = {4,5},

with operations assignment as shown in Table 5.1. The correspon-ding decomposed graph is shown in Figure 5.4.

From the above example, it is obvious that the basic philosophy in use of graph partitioning for generalized grouping case is the same as for the simple grouping. It can further be noticed that this approach, in principle, is different from that of modeling the problem as a p-median problem (or its extension) as followed in the Chapter III.

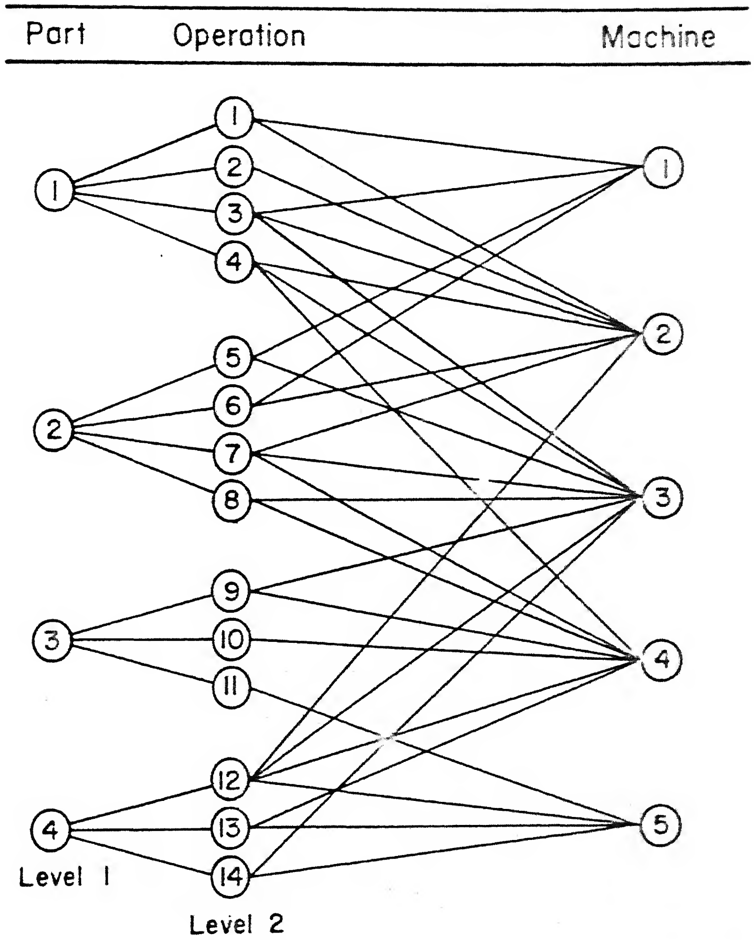


Figure 5.3: AND-OR Graph Representation For a Generalized Grouping Situation (Example 5.2).

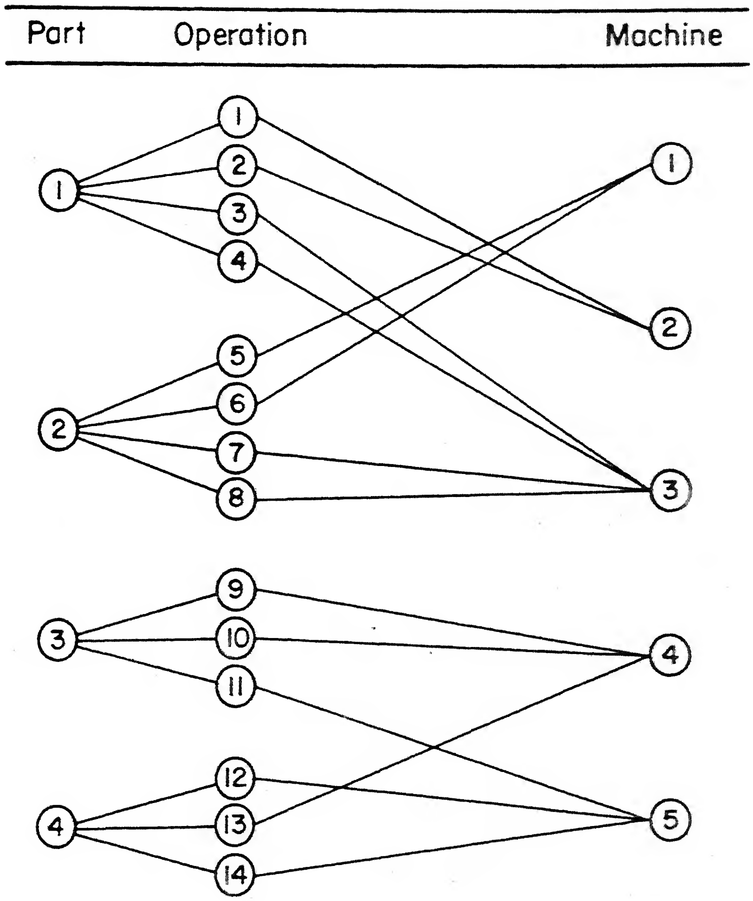


Figure 5.4: Decomposed AND-OR Graph (Example 5.2).

Table 5.1 : Operations Assignment Detail (Example 5.2).

Part	Operation	Machine to which operation is assigned
1	1	2
	2	2
	3	3
	4	3
2	5	1
	6	1
	7	3
	8	3
3	9	4
	10	4
	11	5
4	12	5
	13	4
	14	5

5.4 GROUPING AS AN OPERATIONAL PROBLEM

The idea behind group formation in this situation is to determine group of parts and the corresponding logical groups of machines for a given configuration of manufacturing system. The formulations and the solution methodologies are given in subsequent subsections.

5.4.1 Formulation

The objective functions and the various constraints are described below.

OBJECTIVE FUNCTIONS

(i) Maximization of Association Between Operations of Parts and Machines Assigned to the Same Group

The objective of maximizing the sum of associations between operations of the parts and the machines assigned to the group to

which that part is assigned is represented as:

$$\text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} a_{nim} \sum_{k=1}^{p} x_{nimk}. \qquad (5.1)$$

For the case where (a) there is only one machine of each type, (b) operations are to be performed on only one machine type, and (c) no two operations of a part require the same machine type, the problem of grouping is the same as simple grouping problem, and the objective function and its expression is similar to that stated by Kumar et al (1986) or Ventura et al (1987).

(ii) <u>Minimization of Intercell Movement Cost</u>

If an operation is assigned to some machine then the expression $\sum_{m \in C(i)} \sum_{k=1}^{p} x_{nimk}$ will be equal to one, zero otherwise. Therefore, the expression $(1 - \sum_{m \in C(i)} \sum_{k=1}^{p} x_{nimk})$ will assume value equal to one only when intercell movement is required. Assuming that the notation a_{ij} represents the intercell movement cost related with processing of operation j of part i and is determined considering the production volume and other related relevant factors, the objective of minimizing the intercell movement cost can be expressed as:

$$\text{Minimize} \sum_{n=1}^{N} \sum_{i \in B(n)} a_{ni} \, (1 - \sum_{m \in C(i)} \sum_{k=1}^{p} x_{nimk}).$$

The above expression can be written as:

$$\text{Minimize} \sum_{n=1}^{N} \sum_{i \in B(n)} a_{ni} - \sum_{n=1}^{N} \sum_{i \in B(n)} a_{ni} \sum_{m \in C(i)} \sum_{k=1}^{p} x_{nimk}.$$

In the above expression $\sum_{n=1}^{N} \sum_{i \in B(n)} a_{ni}$ is a constant, thus the objective of minimizing the intercell movement cost is equivalent

(v) Number of Machines Assigned to a Group

From the organizational point of view and also from the view point of material handling, and proper control and supervision, it may be desirable to restrict the number of machines assigned to a group. The constraint on the minimum number of assignments of machines to a group is expressed by (5.8) and that on the maximum number by (5.9).

$$\sum_{m=1}^{M} z_{mk} \geq L_M \qquad k = 1, \ldots, p \tag{5.8}$$

$$\sum_{m=1}^{M} z_{mk} \leq U_M \qquad k = 1, \ldots, p \tag{5.9}$$

(vi) Number of Parts Assigned to a Group

For the similar reasons as mentioned above regarding the restriction on the number of machines that can be assigned to a group, one may like to restrict the number of parts that can be assigned to a group. The limit on the minimum number of parts that must be assigned a group is expressed by (5.10) and that on the maximum number of parts that can be assigned to a group by (5.11).

$$\sum_{n=1}^{N} y_{nk} \geq L_P \qquad k = 1, \ldots, p \tag{5.10}$$

$$\sum_{n=1}^{N} y_{nk} \leq U_P \qquad k = 1, \ldots, p \tag{5.11}$$

(vii) Load on Machines

The constraint that the total load on a machine of the operations that are assigned to it should not exceed the available capacity, can be expressed as:

$$\sum_{n=1}^{N} \sum_{i \in B(n) \cap \underline{B}(m)} t_{nim}\, x_{nimk} \leq T_m z_{mk} \qquad m = 1, \ldots, M; \quad k = 1, \ldots, p \tag{5.12}$$

In the presence of the above constraint, the constraint (5.5) need not be considered.

DECISION VARIABLES

$$x_{nimk} = 0 \text{ or } 1 \qquad n = 1, \ldots, N;\ \forall\, i \in B(n);\ \forall\, m \in C(i);\ k = 1, \ldots, p \qquad (5.13)$$

$$y_{nk} = 0 \text{ or } 1 \qquad n = 1, \ldots, N;\ k = 1, \ldots, p \qquad (5.14)$$

$$Z_{mk} \geq 0 \text{ and integer} \qquad m = 1, \ldots, M;\ k = 1, \ldots, p \qquad (5.15)$$

MODELS

The different combinations of the above mentioned objectives and constraints will encompass different problem scenarios. For example, the problem of only part family determination can be described by the following model.

<u>MODEL M5.1</u>

$$\text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} a_{nim} \sum_{k=1}^{p} x_{nimk}$$

Subject to: (5.3), (5.6), (5.10), (5.11), (5.13) and (5.14).

It can be noted that the above model will simply find out part families and using this result machine cells can be determined later. The model is analogous to the model M3.5(d) developed in the Chapter III based on the approach of p-median formulation. The total number of constraints and the variables required by each of these two formulations are presented in Table 5.2. Since operation i of part n can be performed on any of $|C(i)|$ number of alternative machines, the total number of process plans over all the parts, i.e. q will be:

$$q = \sum_{n=1}^{N} \prod_{i \in B(n)} |C(i)|.$$

From the Table 5.2, it can be seen that for the simple grouping problem, the formulation as for a p-median problem requires lesser number of decision variables and constraints as compared to that for a graph partitioning problem. However, the gap between the requirements of the two approaches starts decreasing with the increase in the number of alternatives for carrying out an operation. In fact, for a real life problem situation such as in FMS environment where for each operation good number of alternatives are available, the formulation using graph partitioning approach will require lesser number of constraints and variables. From Table 5.3, it can be seen that for the Example 5.1 representing a simple grouping situation, the p-median based formulation requires lesser number of decision variables and constraints as compared to graph partitioning based formulation. On the contrary, for the Example 5.2 that represents a generalized grouping situation, the requirement of decision variables is quite high and the number of constraints marginally more. From this observation, it can be concluded that from the view point of the requirements of constraints and decision variables the p-median based formulation will be better than graph partitioning based one when machines are less versatile, and vice versa when machines are more versatile.

The problem of simultaneous group determination without considering the processing times of the operations and machine capacity, but including the constraints on the number of parts and

Table 5.2: Number of Constraints and Variables Required by the Formulations for Determination of only Part Family.

	Model M5.1 (Graph-Partitioning)	Model M3.5(d) (p-median)
<u>Number of Constraints</u>		
(i) when alternative machines exist	$p \sum_{n=1}^{N} \lvert B(n)\rvert + N + 2p$	$2 \sum_{n=1}^{N} \prod_{i \in B(n)} \lvert C(i)\rvert + N + 1$
(ii) when no alternative machine exists	$p \sum_{n=1}^{N} \lvert B(n)\rvert + N + 2p$	$3N + 1$
<u>Number of Variables</u>		
(i) when alternative machines exist	$\sum_{n=1}^{N} \sum_{i \in B(n)} \lvert C(i)\rvert + pN$	$\left\{ \sum_{n=1}^{N} \prod_{i \in B(n)} \lvert C(i)\rvert \right\}^2$
(ii) when no alternative machine exists	$p \sum_{n=1}^{N} \lvert B(n)\rvert + pN$	N^2

Table 5.3: A Comparison Between Requirements of Decision Variables and Constraints for Examples 5.1 and 5.2 for the Formulations Based on p-Median and Graph Partitioning.

	p-median based formulation	Graph partitioning based formulation
Example 1		
(i) Number of variables	25	30
(ii) Number of constraints	16	29
Example 2		
(i) Number of variables	3600	68
(ii) Number of constraints	125	68

machines that can be assigned to a group and the number of machines of each type, can be expressed by the model M5.2 described below.

MODEL M5.2

$$\text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} a_{nim} \sum_{k=1}^{p} x_{nimk}$$

Subject to: (5.3), (5.5), (5.6), (5.7), (5.8), (5.9), (5.10), (5.11), (5.13), (5.14) and (5.15).

The requirements of constraints and variables for this model as compared to the equivalent model presented in the Chapter III are given in Table 5.4.

From the Table 5.4, it can be seen that for the generalized grouping situation graph partitioning approach will, in general, require comparatively lesser number of variables and constraints.

A most generalized situation considering machine capacity constraint and restriction on group disjointedness can be represented by the following model.

MODEL M5.3

$$\text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} a_{nim} \sum_{k=1}^{p} x_{nimk}$$

Subject to: (5.3), (5.6), (5.7), (5.8), (5.9), (5.10), (5.11), (5.12), (5.13), (5.14) and (5.15).

The number of variables and the constraints involved in the two formulations represented by the models M5.2 and M5.3 are the same.

The models M5.2 and M5.3 will not yield feasible solutions if the number of the machines and their processing capacities are not sufficient to result disjoint groups. In such cases, intercell

Table 5.4: Number of Constraints and Variables Required by the Formulations for Simultaneous Determination of Part Families and Machine Cells.

	Model M5.2 (Graph Partitioning)	Model M3.6(b) (p-median)
<u>Number of Constraints</u>		
(i) when alternative machines exist	$p\left[\sum_{n=1}^{N} \lvert B(n)\rvert + M + 4\right] + (N + M)$	$5\sum_{n=1}^{N} \prod_{i\in B(n)} \lvert C(i)\rvert + (N + M + 1)$
(ii) when no alternative machine exists	$p\left[\sum_{n=1}^{N} \lvert B(n)\rvert + M + 4\right] + (N + M)$	$(6N + M + 1)$
<u>Number of Variables</u>		
(i) when alternative machines exist	$p\sum_{n=1}^{N} \sum_{i\in B(n)} \lvert C(i)\rvert + p(N + M)$	$\left\{\sum_{n=1}^{N} \prod_{i\in B(n)} \lvert C(i)\rvert\right\}^2 + M\sum_{n=1}^{N} \prod_{i\in B(n)} \lvert C(i)\rvert$
(ii) when no alternative machine exists	$p\left[\sum_{n=1}^{N} \lvert B(n)\rvert + N + M\right]$	$N^2 + MN$

movements are unavoidable, or to have disjointedness and operationally feasible solutions additional machines or overtime on certain existing machines may be needed. For grouping purpose in these situations, the constraint (5.3) should be replaced by the constraint (5.4) into the models M5.2 and M5.3. The resulting respective models are given below.

MODEL M5.4

$$\text{Maximize} \sum_{n=1}^{N} \sum_{i\in B(n)} \sum_{m\in C(i)} a_{nim} \sum_{k=1}^{p} x_{nimk}$$

Subject to: (5.4), (5.5), (5.6), (5.7), (5.8), (5.9), (5.10), (5.11), (5.13), (5.14) and (5.15).

MODEL M5.5

$$\text{Maximize} \sum_{n=1}^{N} \sum_{i\in B(n)} \sum_{m\in C(i)} a_{nim} \sum_{k=1}^{p} x_{nimk}$$

Subject to: (5.4), (5.6), (5.7), (5.8), (5.9), (5.10), (5.11), (5.12), (5.13), (5.14) and (5.15).

It should be noted that in the models M5.2 and M5.4, the variables Z_{mk}'s can be considered to be binary integer variables if the constraint (5.8) is not included. It should also be noted tht the use of the models M5.4 and M5.5 may result solutions where all the operations may not be performed within the assigned cells though it may be possible to do so. In such cases, the objective function value will generally be higher than for the case of perfect grouping.

As mentioned earlier, the grouping result obtained using the models M5.4 and M5.5 may not result a complete solution for the operations assignment problem. Once the groups of parts and machines have been determined, the problem of operations

assignment can be solved completely using the loading models that will be discussed in the next chapter.

5.4.2 Solution Methodology

For the models of the problem that have been described in the previous section, mathematical programming technique can be used. Some of these problems are relatively simple and can easily be solved while the remaining others are hard to solve. An attempt is made to develop alternative methods for solving these problems in an easier manner. These methodologies for different problem scenarios are described in the following subsections.

5.4.2.1 DETERMINATION OF PART FAMILY ONLY

The problem of determining only part family represented by the model M5.1 can be solved in a very easy way as follows.

First find for each operation of a part, a machine for which corresponding association is the maximum. Let

$$a_{ni\underline{m}} = \underset{m \in C(i)}{\text{maximum}} (a_{nim}).$$

The operation i of part n will be assigned to machine $\underline{m}$. Using this method solution to assignment of other operations can be found. The constraint on the maximum and the minimum number of parts to be assigned to a group is trivial and can easily be satisfied. The parts can be assigned to any of the p groups but ensuring that each group gets at least L_p parts and no more than U_p parts.

It can be observed from the above methodology for part family determination that the parts that may get assigned a group need not be close to each other in terms of their machine requirements.

variables w_{mk}'s the subproblem P_a is effectively the same as the problem of only part family determination (model M5.1) where in the objective function expression the term a_{nim} is replaced by $(a_{nim} - w_{mk})$. Thus, the methodology described in the Section 5.4.2.1 can be used accordingly for solving the problem P_a.

Further, for the known values of w_{mk}'s the problem P_b will bear the structure of a transportation problem as can be seen from the following.

Introducing slack variables s_m, u_k and v_k into the respective constraints (5.7), (5.8) and (5.9), the following equations are obtained.

$$\sum_{k=1}^{p} Z_{mk} + s_m = |N_m| \qquad m = 1, \ldots, M \tag{5.17}$$

$$\sum_{m=1}^{M} Z_{mk} - u_k = M_L \qquad k = 1, \ldots, p \tag{5.18}$$

$$\sum_{m=1}^{M} Z_{mk} + v_k = M_U \qquad k = 1, \ldots, p \tag{5.19}$$

where
$$s_m \geq 0 \text{ and integer} \qquad m = 1, \ldots, M \tag{5.20}$$
$$u_k,\ v_k \geq 0 \text{ and integer} \qquad k = 1, \ldots, p \tag{5.21}$$

From the equations (5.18) and (5.19),

$$u_k + v_k = M_U - M_L \qquad k = 1, \ldots, p. \tag{5.22}$$

Summing the equation (5.17) over m and then subtracting from it the equation (5.18) after summing it over k, the following constraint is obtained.

$$\sum_{m=1}^{M} s_m + \sum_{k=1}^{p} u_k = \sum_{m=1}^{M} |N_m| - p\, M_L \tag{5.23}$$

The problem P_b can be restated as:

$$\text{Maximize} \sum_{m=1}^{M} \sum_{k=1}^{p} w_{mk} Z_{mk}$$

Subject to: (5.17), (5.19), (5.22), (5.23), (5.15), (5.20) and (5.21).

The transportation matrix corresponding to the above formulation is shown in Figure 5.5. The rows 1 to M represent the constraint (5.17), whereas that from (M+1) to (M+p) the constraint (5.22). The columns 1 to p represent the constraint (5.19), whereas the column (p+1) the constraint (5.23).

The objective function value $Z_D(w)$ of the relaxed problem is an upper bound on the objective function value of the original problem. Therefore, dual variables w^* determined should make the upper bound $Z_D(w)$ as small as possible. This problem can be solved using subgradient optimization procedure suggested by Held et al (1974).

Let the value of the objective function in the model M5.4 be Z_P. The various steps of the subgradient procedure for the present problem are as follows.

Step 0: Set $j = 1$, $w^j = 0$, $\lambda^j = \lambda^o$, $LB^o = 0$ and

$$UB^o = \sum_{n=1}^{N} \sum_{i \in B(n)} \underset{m \in C(n,i)}{\text{maximum}} (a_{nim})$$

where,

w^j : value of dual variables w_{mk}'s at iteration j

LB^o : initial lower bound

	1		k		p	p+1	
1	w_{11} Z_{11}		w_{1k} Z_{1k}		w_{1p} Z_{1p}	0 s_1	$\lvert N_1\rvert$
m	w_{m1} Z_{m1}		w_{mk} Z_{mk}		w_{mp} Z_{mp}	0 s_m	$\lvert N_m\rvert$
M	w_{M1} Z_{M1}		w_{Mk} Z_{Mk}		w_{Mp} Z_{Mp}	0 s_M	$\lvert N_M\rvert$
M+1	0 v_1		*		*	0 u_1	M_U-M_L
M+k	*		0 v_k		*	0 u_k	M_U-M_L
M+p	*		*		0 v_p	0 u_p	M_U-M_L
	M_U		M_U		M_U	$\sum_{m=1}^{M} \lvert N_m\rvert - k\,M_L$	

Figure 5.5: Transportation Matrix Corresponding to Subproblem P_b.

UB^o : initial upper bound

λ^j : value of the scalar at iteration j ($0.0 \le \lambda^o \le 2.0$).

Step 1: Solve the problem P_a and P_b for the current set of Lagrange multipliers w^j. Let x^j, y^j and z^j be the optimal solution, and the objective function values of the problems P_a and P_b be Z_1^j and Z_2^j, respectively. Set $Z_D^j = Z_1^j + Z_2^j$.

Step 2: Update the upper bound:

$UB^j = \text{minimum}\ (UB^{j-1},\ Z_D^j)$.

Step 3: If x^j, y^j and z^j are feasible to the primal problem, set $Z_P^j = Z_p\ (x^j,\ y^j,\ z^j)$ and go to the next step.

Otherwise, modify x^j in view of y^j and z^j to make it feasible. Let the modified value of x_{nimk}'s be denoted by vector $\underline{x}^j$. Set $Z_P^j = Z_P(\underline{x}^j,\ y^j,\ z^j)$ and go to the next step.

Step 4: Update the lower bound:

$LB^j = \text{maximum}\ (LB^{j-1},\ Z_P^j)$.

Step 5: Determine the value of λ^j. If $\lambda^j < \epsilon_1$, stop.

Step 6: If $(UB^j - LB^j)/LB^j < \epsilon_2$, then stop. Otherwise, go to the next step.

Step 7: Compute the subgradients:

$$S_{mk}^j = \sum_{n=1}^{N} \sum_{i \in B(n) \cap \underline{B}(m)} x_{nimk}^j - L\ Z_{mk}^j \qquad m = 1, \ldots, M; \quad k = 1, \ldots, p.$$

Step 8: Compute the step size:

$$f^j = \frac{\lambda^j\ [Z_P^j - LB^j]}{(\sum_{m=1}^{M} \sum_{k=1}^{p} S_{mk}^j)^2}.$$

Step 9: Update the Lagrangian multipliers:

$$w_{mk}^{j+1} = \text{maximum}\ \{0,\ w_{mk}^{j} + f^{j}\ s_{mk}^{j}\} \qquad m = 1, \ldots, M; \quad k = 1, \ldots, p.$$

Set j = j+1 and go to the Step 1.

The value of ϵ_1 and ϵ_2 are to be fixed suitably depending upon the kind of convergence required. The details on the reduction of the value of λ^j can be had from the work of Held et al (1974).

If at the Step 3 of the subgradient algorithm a feasible solution is not obtained, then the following procedure is used to modify the values of x_{nimk}'s to make them feasible.

$$\underline{x}_{nimk}^{j} = \begin{cases} 1 & \text{if } y_{nk}^{j} = 1 \text{ and } a_{nim} > 0 \text{ where,} \\ & a_{nim} = \underset{\underline{m} \in C(i)}{\text{maximum}}\ (a_{ni\underline{m}} . Z_{\underline{m}k}) \\ 0 & \text{otherwise} \end{cases}$$

In use of the above procedure, some of the operations may remain unassigned. However, solution resulted is feasible because of its conformance with the formulation and the problem requirement as can be seen from the constraints (5.4) and (5.6).

From the above, it can be observed that for the problem of simultaneous group determination where all the operations of a part are to be assigned to some machines that belong to the same group (i.e. for the problem represented by the model M5.2), it will not be possible to determine a feasible solution to the primal problem using the method described above. Even determination of a feasible solution, otherwise also, may not be a simple task. In view of this it can be concluded that for both of the

models M5.2 and M5.3, a heuristic approach based on Lagrangian relaxation of the kind described above cannot be used.

5.4.2.3 SIMULTANEOUS DETERMINATION OF GROUPS CONSIDERING MACHINE CAPACITY

The problem of simultaneous determination of groups of parts and machines considering machine capacity is the same as the problem of simultaneous group determination without considering machine capacity when the processing times of the operations on the various machines are one unit of time and the capacities of the machines are equal to L units of time. Since an instance of the model M5.5 (and the model M5.3) is equivalent to the model M5.4 (and the model M5.2, respectively) and the model M5.4 is stated to be an NP-complete problem in the earlier section, the model M5.5 will also be NP-complete. For the reasons as mentioned in the Section 5.4.2.2, a heuristic approach for the model M5.5 may be desired.

In this section, a heuristic approach based on Lagrangian Relaxation is proposed for the model M5.5. On dualizing the constraints (5.12) and rearranging the terms in the objective function, the following is obtained.

(D_2)

$$Z_D(w) = \text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} \sum_{k=1}^{p} (a_{nim} - w_{mk}\, t_{nim}) x_{nimk}$$

$$+ \sum_{m=1}^{M} \sum_{k=1}^{p} T_m\, w_{mk}\, z_{mk}$$

Subject to: (5.4), (5.6), (5.7), (5.8), (5.9), (5.10), (5.11), (5.13), (5.14) (5.15) and (5.16).

In the above formulation, the constraint (5.12) is dualized using dual variables w_{mk}'s. The problem D_2 for known values of w_{mk}'s can again be decomposed into two subproblems P'_a and P'_b as shown below.

(P'_a)

$$Z'_1 = \text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} \sum_{k=1}^{p} (a_{nim} - w_{mk}\, t_{nim}) x_{nimk}$$

Subject to: (5.4), (5.6), (5.10), (5.11), (5.13) and (5.14).

(P'_b)

$$Z'_2 = \text{Maximize} \sum_{m=1}^{M} \sum_{k=1}^{p} T_m\, w_{mk}\, Z_{mk}$$

Subject to: (5.7), (5.8), (5.9) and (5.15).

It can be noticed that the problems P'_a and P'_b are similar to P_a and P_b, respectively. Therefore, the methodology used for solving the problems P_a and P_b can be used to solve the problems P'_a and P'_b, respectively. The subgradient algorithm to be used for solving the grouping problem will be the same as described in the Section 5.4.2.2. However, Z_P and Z_D in the algorithm will be the objective function values of models M5.5 and D_2, respectively. However, for finding feasible solution to the primal problem M5.5 at Step 3 of the algorithm, a different method as described below will be used.

Let $G_P(k)$ and $G_M(k)$ denote k^{th} group of parts and machines, respectively. To get a primal feasible solution, the operation assignment problem can be solved for each group independently. The corresponding formulation for the k^{th} group will be:

$$\text{Maximize} \quad \sum_{n\in G_P(k)} \sum_{i\in B(n)} \sum_{m\in C(i)\cap G_M(k)} a_{nim}\cdot \underline{x}_{nimk} \tag{5.24}$$

Subject to:

$$\sum_{m\in C(i)\cap G_M(k)} \underline{x}_{nimk} \le 1 \qquad \forall\, n \in G_P(k)$$

$$\forall\, i \in B(n) \tag{5.25}$$

$$\sum_{m\in G_P(k)} \sum_{i\in B(n)\cap \underline{B}(m)} t_{nim}\, \underline{x}_{nimk} \le T_m\, Z_{mk}$$

$$\forall\, m \in G_M(k) \tag{5.26}$$

The above problem has the structure of a generalized assignment problem and thus is NP-Complete. For any feasible solution to the above problem, the objective function value will be at most equal to the optimal objective function value. The problem as described in the above formulation happen to be the part of the primal problem and thus the sum of their objective function values will be equal to the objective function value of the primal problem. Hence, the sum will provide a lower bound on the value of primal objective. Since the lower bound is to be kept at the maximum, finding solution to the above operation assignment problem using some heuristic will not be advantageous but may require lesser computational effort. Thus, a tradeoff has to be made between higher objective function value and the computational ease. An optimal solution to the above problem can be found using branch and bound approach suggested by Ross and Soland (1975) or dual-based algorithm of Guignard and Rosenwein (1989). Since the computational requirement of an integer programming problem increases, in general, with the increase in the problem size, thus to reduce the computational burden certain decision variables can be removed from the consideration using the procedure as follows.

part $i \in P$, the sum of the associations between the operations of the part and the number of different machine types used in that process plan. Denote the maximum sum (to be called as gain) by G_i and the corresponding process plan by PR_i.

Sequence all the parts in descending order of the number of machines used by the parts in the process plans corresponding to which their gains are maximum. Resolve the tie based on the value of G_i's. Further ties can be broken arbitrarily.

Set $U = \phi$, $K = 1$ and go to the next step.

Step 1: Denote the first part in the sequence by p. Put the part p into part family PF_K and all the machine types in PR_p to form machine cell MC_K. Put into part family PF_K also those parts whose association with the machines in MC_K is more than half of their maximum gain.

Reduce the number of copies available for the machine types in MC_K by one. Remove the parts in PF_K from the sequence and go to the next step.

Step 2: If all the parts have been considered for assignment to the groups, then follow Phase II. Otherwise, go to the next step.

Step 3: Find the gain of first part (let the part be p) in the sequence for each of the machine cell v ($v = 1, ..., K$) considering all its alternative process plans. Denote the machine cell by k and the process plan by PR'_p for which the gain of the part p, i.e. G'_p, is maximum. If the gain G'_p is more than $G_p/2$, assign the part p to k-th

part family and remove it from the sequence, and go to the Step 2. Otherwise, follow the next step.

Step 4: Find for the part p, the maximum gain G'_p considering only the unassigned machines and all its alternative process plans. Let G'_p be maximum corresponding to process plan PR'_p. If G'_p is more than $G_p/2$, then form a new group and set $K = K+1$. The K-th machine cell (MC_K) will have all the machine types that are used by the process plan PR'_p and of which some copy is still available for assignment. Reduce the number of copies available of the machine types in MC_K by one. The part p is assigned to part family PF_K.

If G'_p is not greater than $G_p/2$, put part p in set U and go to the Step 2.

<u>Phase II</u>

Step 0: Let V denote the set of machine types whose some copies are still unassigned. If U or V are nonempty, then set $K = K + 1$. The parts in the set U and the machines of the type as in the set V will form K-th group.

Step 1: Consider each of the machines belonging to each of the machine cells for possible placement to a different cell. A machine will be assigned to a different cell only when it brings improvement in the objective function value, and to the cell for which the improvement is maximum.

Step 2: Similarly, consider the parts for possible placement to different families. A part will be assigned to a different family only when it brings improvement in the objective function value and to the family for which the

improvement is maximum.

Step 3: Repeat Steps 1 and 2 until no reassignment of machines and parts takes place.

Total number of groups formed may be less than K because of the adjustments made in Phase II and also because of the exclusion of those groups from consideration to which no part is assigned. It should be noted that while determining the improvement in objective function value in the Step 1 of Phase II of the heuristic, the contribution of machine is taken to be zero for all those groups that already have another copy of the same machine type.

The flowcharts for the two phases of the heuristic are given in Figures 5.6 and 5.7.

<u>Finite Convergence of the Heuristic</u>

The heuristic proposed will terminate in finite number of steps. The finiteness of the Phase I is self evident and thus is not explained. In Phase II, the parts and machines keep shifting from one group to another. Since each shift definitely brings improvement in the objective function value and the maximum value of the objective function is also finite, the procedure will terminate in finite number of steps.

5.4.3.2 SIMPLE GROUPING PROBLEM

The simple grouping problem is a special case of generalized problem, and thus the heuristic described in the Section 5.4.3.1 can also be used for it. However, the simplicity of the problem can be exploited to bring improvement in the objective function value by suitably modifying the heuristic described in the Section 5.4.3.1. The problem considered for the grouping is more general

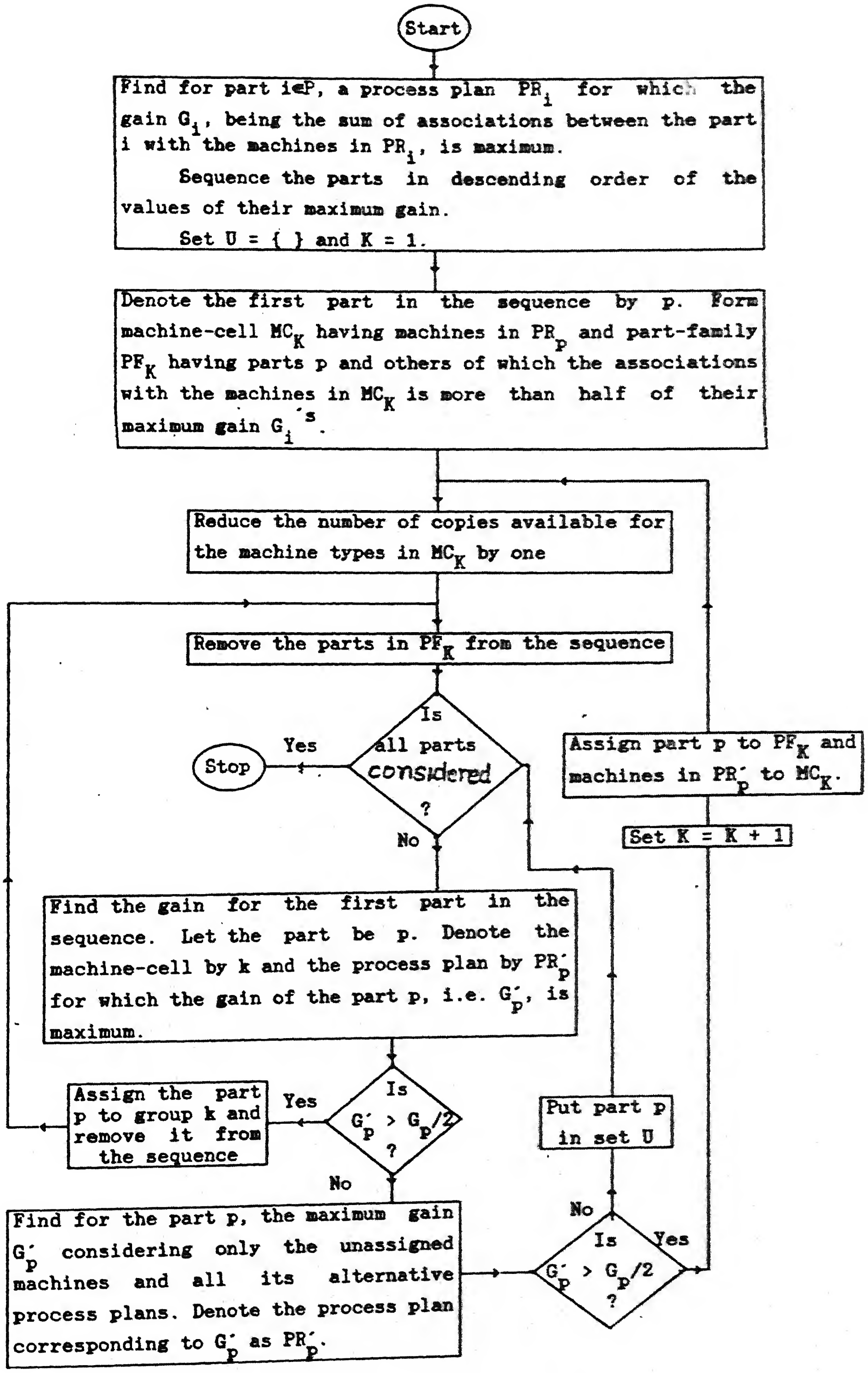


Figure 5.6: Flowchart Corresponding to Phase I of the Heuristic for the Generalized Grouping.

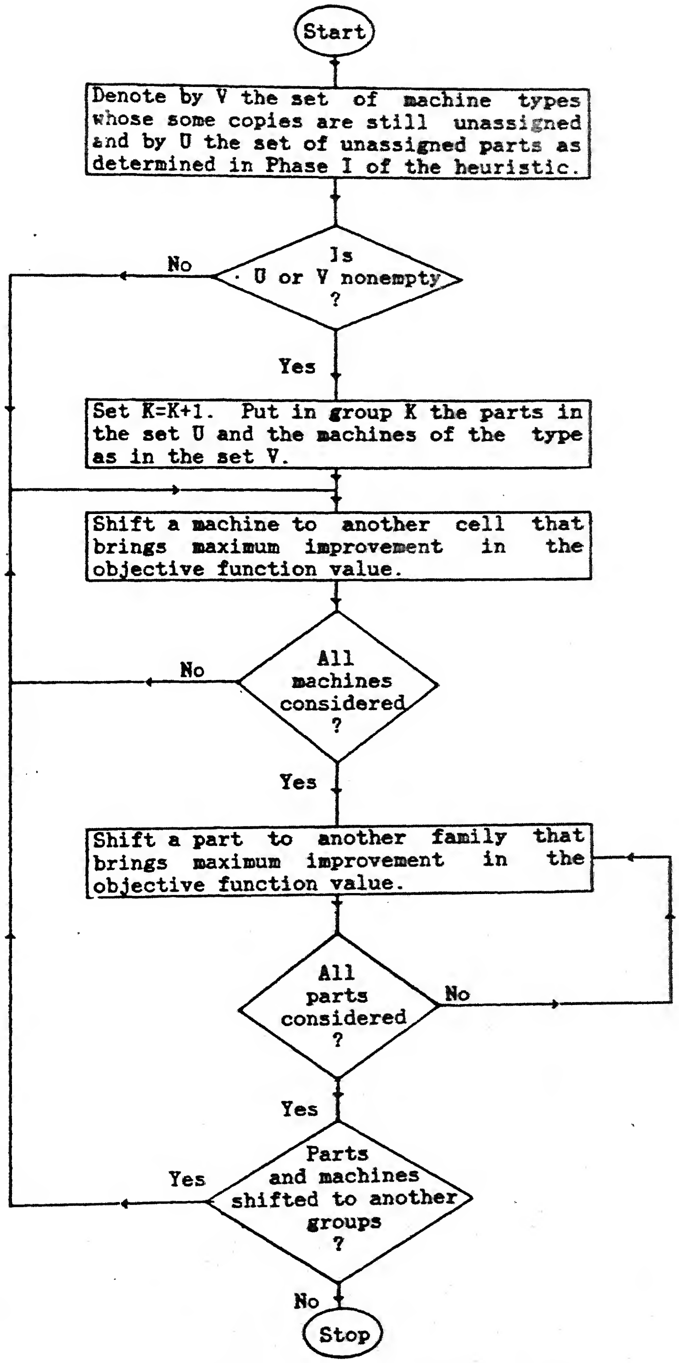


Figure 5.7: Flowchart Corresponding to Phase II of the the Heuristic for the Generalized Grouping.

and considers the restrictions on group size and number of groups to be formed.

In simple grouping situation, the operations are to be performed on only one machine. Thus, the operations that require the same machine type can be merged together and so their associations with machines, processing times and costs. For a situation like this, maintaining separate indices for the operations has no significance and thus can be omitted. The problem can be represented by the following formulation.

$$\text{Maximize} \sum_{n=1}^{N} \sum_{m \in C(n)} a_{nm} \sum_{k=1}^{p} x_{nmk}$$

Subject to:

$$x_{nmk} \le y_{nk} \qquad n = 1, \ldots, N; \ \forall\, m \in C(n); \ k = 1, \ldots, p$$

$$\sum_{n \in \underline{C}(m)} x_{nmk} \le Z_{mk} \qquad m = 1, \ldots, M; \ k = 1, \ldots, p$$

$$\sum_{k=1}^{p} y_{nk} = 1 \qquad n = 1, \ldots, N$$

$$\sum_{k=1}^{p} Z_{mk} \le |N_m| \qquad m = 1, \ldots, M$$

$$P_L \le \sum_{n=1}^{N} y_{nk} \le P_U \qquad k = 1, \ldots, p$$

$$M_L \le \sum_{n=1}^{M} Z_{mk} \le M_U \qquad k = 1, \ldots, p$$

$$x_{nmk} = 0 \text{ or } 1, \qquad n = 1, \ldots, N; \ \forall\, m \in C(n); \ k = 1, \ldots, p$$

$$y_{nk} = 0 \text{ or } 1 \qquad n = 1, \ldots, N; \; k = 1, \ldots, p$$

$$Z_{mk} \geq 0 \text{ and integer} \qquad m = 1, \ldots, M; \; k = 1, \ldots, p$$

In the above formulation, various notations (though simplified) have their usual definitions and meaning. The notation C(n) represents the set of machine types required by part n, $\underline{C}(m)$ represents the set of parts that need machine type m. The value of association between a part n and machine type m, i.e. a_{nm} is also termed as requirement of part n on machine type m. In case when there is no restriction on the minimum number of parts to be assigned to a group, variables Z_{mk}'s may be binary instead of being general integer variables.

The heuristic proposed for the above problem uses two phase procedure for group formation. The first phase determines natural groups, i.e. the groups that contain parts and machines which are closely related. In the natural groups, most of the parts should meet at least half of their total requirements from the machines assigned to the groups to which the parts are assigned. In Phase II, the groups formed are adjusted to satisfy the restrictions on the number of groups to be formed and size of the groups, and further the groups are refined to bring improvement in the objective function value.

The heuristic presented is quite general and can be used for finding solution of the grouping problem with no restrictions as mentioned below or with any combination of these:

(i) Number of groups to be formed.

(ii) Minimum number of parts in a group.

(iii) Maximum number of parts in a group.

(iv) Minimum number of machines in a group.

(v) Maximum number of machines in a group.

It should be noted that even when there is no restriction on the number of groups to be formed, the limits on the size of part families and machine cells may indirectly control the number of groups to be formed. Hence, the limit on the group size should be imposed carefully. Let

$$L_1 = N/P_L$$

$$L_2 = N/P_U,$$

$$L_3 = \sum_{m=1}^{M} |N_m|/M_L,$$

and

$$L_4 = \sum_{m=1}^{M} |N_m|/M_U,$$

Then the minimum number of groups (p_{min}) that can get formed will be equal to maximum of L_2 and L_4, and the maximum number of groups (p_{max}) that can get formed will be equal to minimum of L_1 and L_3. Thus, a valid number of groups p to be formed must satisfy the following inequality:

$$p_{min} \le p \le p_{max}.$$

In case where all the limits on the number of parts and machines in a group are not specified, the lower and upper bounds on the number of groups can be determined by fixing the limits not specified as follows.

Minimum number of parts in a group = 1

Maximum number of parts in a group = N

Minimum number of machines in a group = 1

parts and machines for group formation. If still the procedure fails to form a group, conclude that the problem specified does not contain natural groups and follow Phase II.

Step 5: Denote the set of remaining parts and machine types by P' and M'. Set K = K+1. If P' and M' are non-empty, then sequence the parts in P' in descending order of the number of machine types used that are present in M' and go to the next step, else go to Step 9.

Step 6: The first part in the sequence is chosen to form K-th group. The K-th machine cell will have all those machine types that are used by this part and are available for assignment (i.e. member of set M'). The K-th part family will have this part and all other unassigned parts whose more than half requirement can be met by this newly formed machine cell.

Step 7: From the group formed, remove the parts and machines using the method as described in Step 2.

If both, the K-th machine cell and the part family are not empty, go to Step 5. Otherwise, follow, the next step.

Step 8: Take out this part from the sequence. If there is no part in the sequence left for consideration, go to the next step. Otherwise, make the K-th group empty and go to the Step 6.

Step 9: Assign to K-th group all the parts in P' and unassigned machines of the type as in M'.

Now consider the machines assigned to the various groups for possible placement into different machine cells. A machine will be assigned to a different cell only when it brings improvement in the objective function value and to the cell for which the improvement is maximum. Similarly, consider the parts assigned to the various groups for possible placement into different families. A part will be assigned to a different family only when it brings improvement in the objective function value and to the family for which this improvement is maximum. Repeat this process until no shift takes place.

Step 10: If the last group (i.e. the group K) has collapsed, then set K = K-1 and follow Phase II. Otherwise, find disjoint groups of the parts and machines from the parts and machines in group K, and put them in the list of the groups already formed. Let the number of disjoint groups formed from group K is k, then the total number of groups will be equal to K+k-1. Set K = K + k - 1 and follow Phase II.

<u>Phase II</u>

Step 1: Determine p_{min} and p_{max}. If the number of groups to be formed has not been specified, then go to Step 1 else to Step 3.

Step 2: If $p_{min} \leq K \leq p_{max}$, then set $p = K$ and go to the next step.

Otherwise, if $K < p_{min}$, set $p = p_{min}$ and go to the

next step. In the other condition where $K > p_{max}$, set $p = p_{max}$ and go to the next step.

Step 3: If $p < K$, then merge $(K - p)$ groups with other groups bringing maximum improvement in the objective function value. Otherwise create $(p-K)$ new groups in a way that requires minimum reduction in the objective function value. Go to Step 4.

Step 4: Shift the parts and machines from one group to another to satisfy the limits on the size of part family and machine cell in a way that brings least reduction in the objective function value.

Step 5: Readjust the groups by shifting parts and machines from one group to another to bring improvement in the objective value. Adjustments made should satisfy the limits on the size of part family and machine cell.

Continue this process until no shift takes place and then stop.

The heuristic, as can be seen, does not try to order the rows (parts) and columns (machines) but assigns a part (or a machine) to a group where its requirements are maximally satisfied. For this reason, the heuristic will be called as Requirement Based Clustering (RBC) methodology. The characteristic of the procedure RBC of forming the groups in a natural way can be observed from the steps 1, 2, 6 and 7 of the heuristic. The flowcharts for the Phase I and Phase II of the heuristic are given in Figures 5.8 and 5.9, respectively.

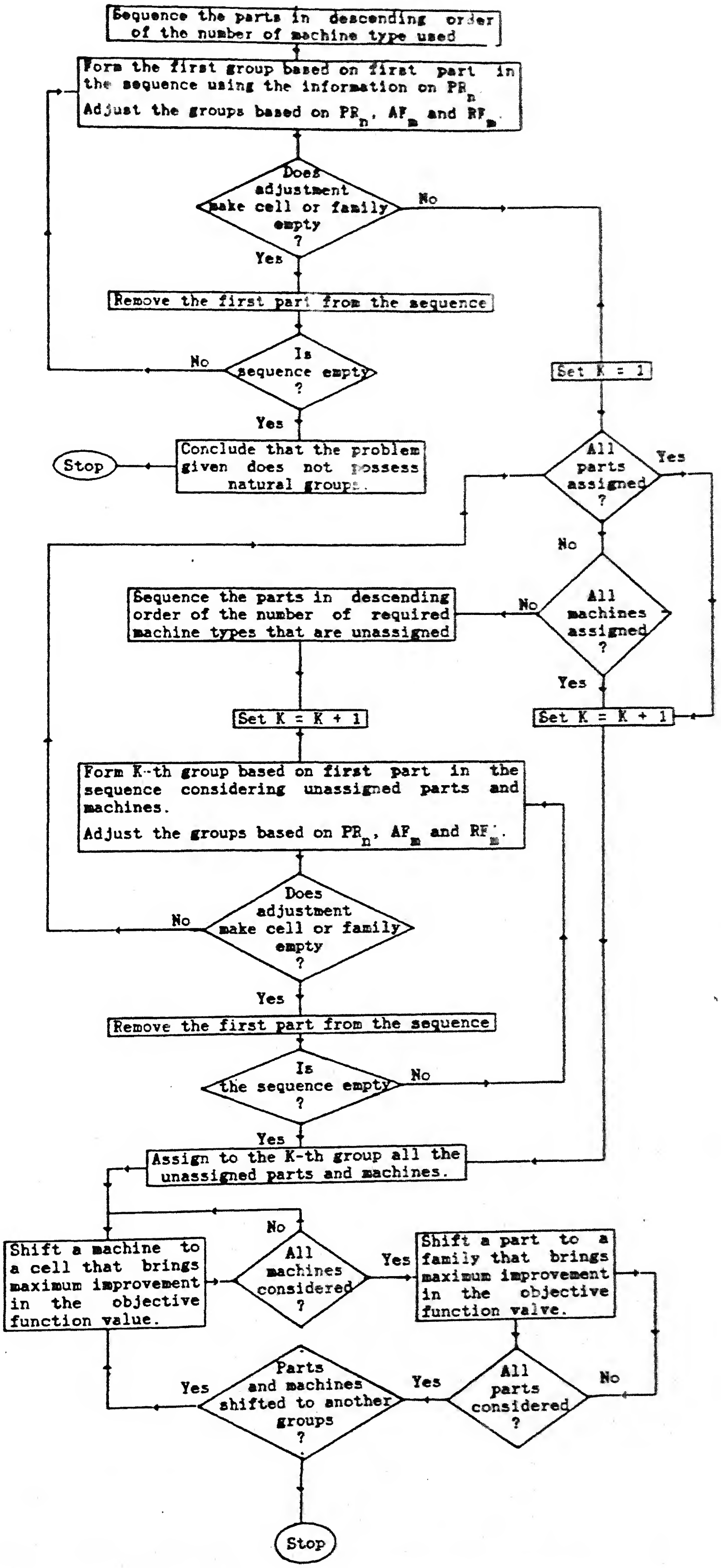


Figure 5.8: Flowchart Corresponding to Phase I of the the Heuristic for the Simple Grouping.

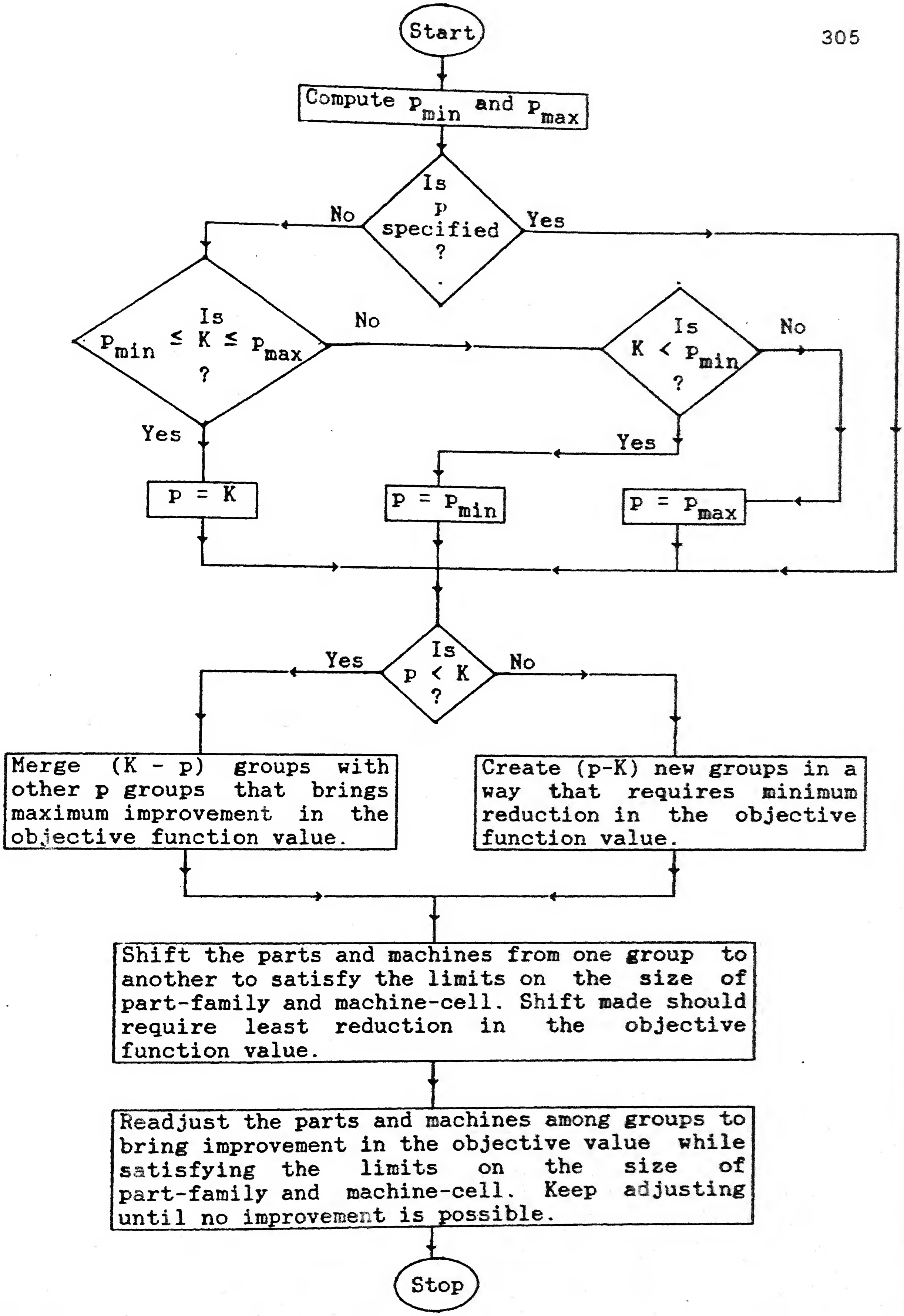


Figure 5.9: Flowchart for the Phase II of the Heuristic for the Simple Grouping Problem.

For finding the disjoint groups at Step 10 in Phase I of the heuristic, the algorithm suggested by Deo (1989) for finding disconnected component of a graph can be used. A flowchart of this algorithm is shown in Figure 5.10. The Step 5 of the Phase II in the same as the Step 9 of the Phase I except that in this step precaution is taken to avoid violation of the limits on size of part families and machine cells. The details of the methods used for merging the existing groups and for creating the new ones are given in the flowcharts shown in Figures 5.11 and 5.12, respectively. The method used in the Step 4 of Phase II for satisfying the limits on group size is described in the flowchart given in Figure 5.13.

The heuristic proposed has been successfully used by Saraf (1989) where for a problem with more than 1000 components and 75 assemblies (that is equivalent to machines), it took less than a minute of CPU time on DEC-1090 computer to find natural groups.

The heuristic RBC can directly use the processing time requirements of parts on different machine types for group formation. However, the assignment of machines to a group based on its capacity and processing time requirements of the parts will require little modification in the heuristic that can easily be incorporated.

The heuristic RBC seems to be quite general as compared to within-cell utilization based heuristic (WUBC) proposed by Ballakur and Studel (1987) in terms of the factors that can be used as the basis for group formation and the restrictions on the number of groups, the minimum and the maximum number of parts and

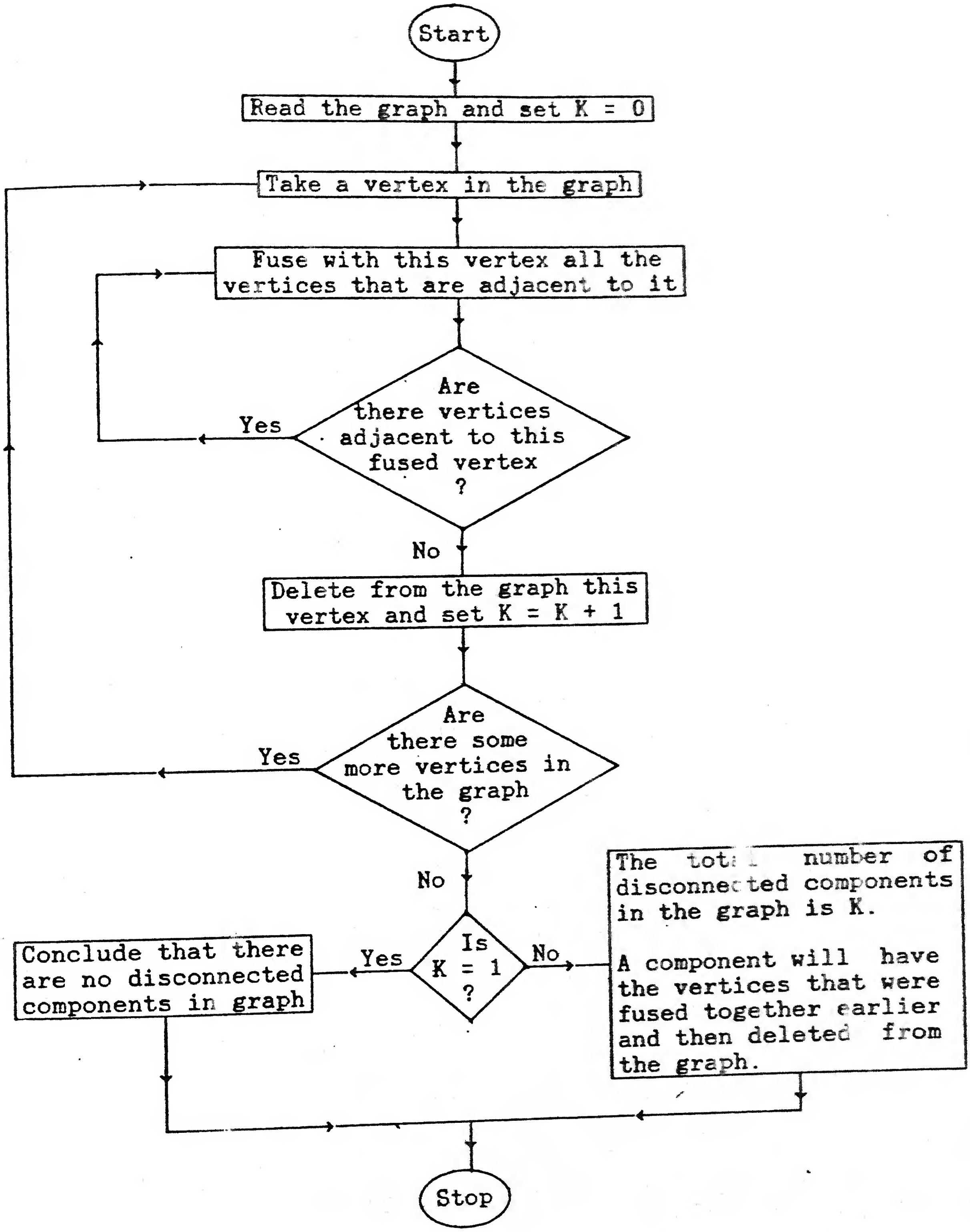


Figure 5.10: Flowchart of the Algorithm Used for Finding Disconnected Components of a Graph.

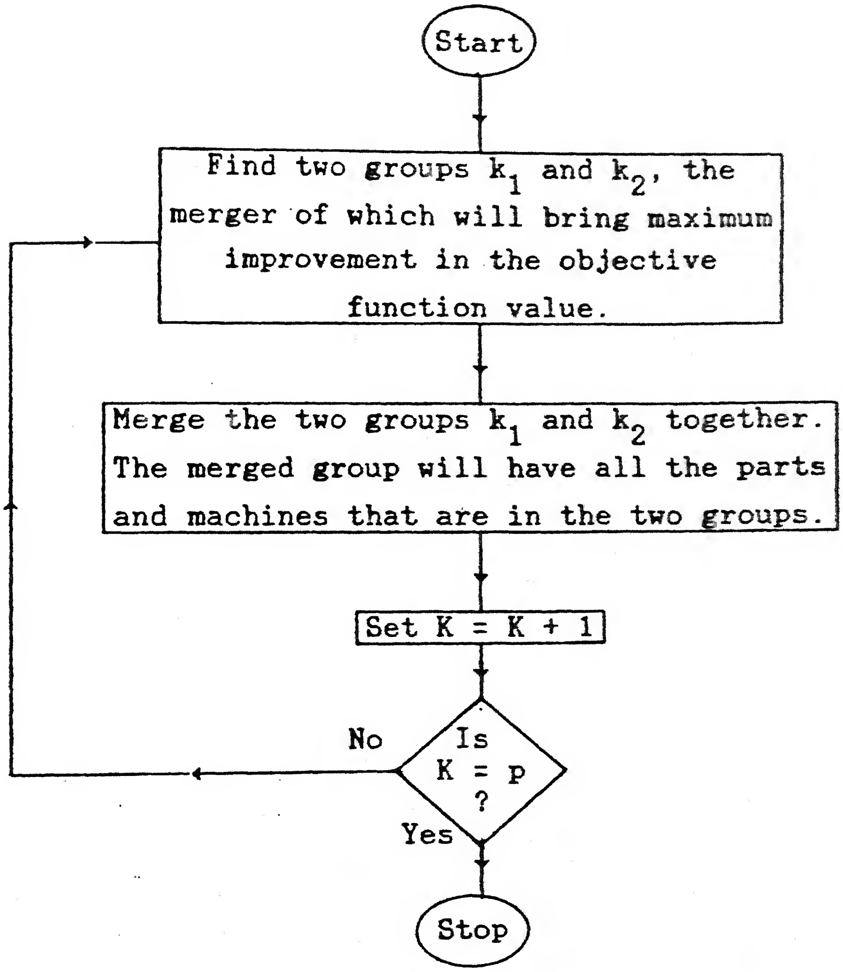


Figure 5.11: Flowchart Showing the Method Used for Obtaining Given Number of Groups in the Simple Grouping Situations When the Total Number of Groups Formed are More than What is Required.

machines in the groups. The heuristic RBC results natural groups where a part stays in a group where its requirements are met maximally, and the machine in a group where it is required most. However, the heuristic WUBC fails to do so because machines once assigned to a cell based on cell-admissibility factor are discarded from the consideration of their placement to other groups where they may be required comparatively more. Moreover, the heuristic WUBC does not try to reassign parts and the machines in the remainder group to possibly result a better grouping.

The flexible grouping technique (FGT) proposed by Dao and Boisclair (1989) includes some of the basic features used in the RBC procedure. However, the proposed method is much more general and is supposed to yield better grouping. The technique FGT does not consider any of the restrictions that can be handled by RBC procedure. Further, it may not yield natural groups because of its method used for group formation which is similar to that based on cell-admissibility-factor of the heuristic WUBC and also because of its lackness for reconsideration of parts and machines already assigned for placement to other groups.

The problems reported in the literature have been solved using the RBC procedure and the solutions obtained are better or the same in terms of the number of exceptional elements.

5.5 GROUPING AS A PLANNING PROBLEM

The scope of grouping at the planning level is different from that at the operational level. As mentioned earlier, grouping at the operational level is generally logical, while that at the planning level generally physical. At the planning level, GT concepts can be used in designing the production system and

determining the number of machines of each type and their configuration in each of the manufacturing cell. Such an analysis and decision making will be justifiable for dedicated systems where the parts to be produced and their characteristics are known in advance and are not going to change for some reasonably good amount of time.

The formulations of the grouping problems are presented below. Comments on certain aspects of the grouping consideration are also given.

5.5.1 Formulation

The constraints considered and their expressions are the same as given in the Section 5.4.1. However, $|N_m|$ will represent the maximum number of machines of type m that can be put into the system. The objectives considered for the grouping are different from that given in the Section 5.4.1 and is the minimization of the total invesment on machines. In the resulting cellular manufacturing system, the groups work independently and intercell movements do not result. The objective of minimizing the total investment on machines can be expressed as:

$$\text{Minimize} \sum_{m=1}^{M} f_m \sum_{k=1}^{p} Z_{mk}. \tag{5.27}$$

For a right kind of analysis, the cost of machines (i.e. f_m) should be considered properly taking into account, for example, the planning period within which all the parts are to be manufactured.

The different combination of the constraints will capture different grouping scenarios. However, for the illustration the models are presented only for the most generalized grouping

situations and are as given below.

MODEL M5.6

$$\text{Minimize} \sum_{m=1}^{M} f_m \sum_{k=1}^{p} Z_{mk}$$

Subject to: (5.3), (5.6), (5.7), (5.8), (5.9), (5.10), (5.11), (5.12), (5.13), (5.14) and (5.15).

Because of the reasons as mentioned in the previous chapter, it may be desired to restrict the total number of machines in the system. For this situation, the following constraint should be included into the models M5.6 and M5.7.

$$\sum_{m=1}^{M} \sum_{k=1}^{p} Z_{mk} = M_T \tag{5.28}$$

5.5.2 Solution Methodology

The problem can be solved using mathematical programming techniques, but because of the NP-complete structure it may not be desired. A heuristic approach based on Lagrangian relaxation similar to that as described in the Section 5.4.2.3 will have the problem of determining a feasible solution to the primal problem in the use of subgradient algorithm and thus cannot be used.

5.6 NUMERICAL EXAMPLES

In this section, few examples are given to illustrate the use of the models and some of the heuristic procedures described in the previous sections. These examples also help in better understanding of different grouping situations. The problems posed as mathematical programming problems are solved on IBM compatible PC/AT using LINDO software unless mentioned specifically.

Consider a manufacturing situation with 6 parts and 6 machines to be partitioned into two groups. The maximum and the

minimum number of parts in a group are to be 4 and 2, respectively. The same restriction is also to be observed for the size of the machine cells. The other details of the problem are given in Tables 5.5 and 5.6.

Table 5.5: Processing Requirement and Productivity Potential Details of the Operations of the Various Parts Associated with the Compatible Machines.

Part (n)	Operation ($i \in B(n)$)	Required Machine ($m \in C(n,i)$)	Processing Time Requirement (t_{nim})	Productivity Potential (a_{nim})
1	1	1	1	5
	2	2	1	2
	3	3	1	5
		4	1	3
2	4	1	1	7
		2	1	5
	5	2	1	8
		6	1	20
3	6	1	1	5
		2	1	6
	7	2	1	4
		3	1	5
4	8	3	1	2
		4	1	6
	9	5	1	9
	10	2	1	5
		6	1	10
5	11	1	1	6
		4	1	7
	12	5	1	8
6	13	4	1	4
	14	2	1	3
		5	1	5

Table 5.6: Details of the Machines.

Machine Type	Number of Copies	Capacity of Each Machine	Cost of Each Machine	Set of Compatible Operations
1	1	3	11	1,4,6,11
2	1	3	12	2,4,5,6,7,10,14
3	1	1	14	3,7,8
4	1	3	13	3,8,11,13
5	1	4	10	9,12,14
6	1	1	15	5,10

Table 5.7: Operations Assignment Detail for the Case of Only Part Family Determination.

Objective function value = 99

Part	Operation	Assigned Machine
1	1	1
	2	2
	3	3
2	4	1
	5	6
3	6	2
	7	3
4	8	4
	9	5
	10	6
5	11	4
	12	5
6	13	4
	14	5

In case of only part family determination, i.e. for the problem described by the model M5.1 solution obtained is as shown in Table 5.7. One of the possible sets of groups of the parts is as shown below.

Part-family # 1 = {1, 2, 3}

Part-family # 2 = {4, 5, 6}

Table 5.8: Operation Assignment Details for the Case When the Operations are Carried Out Within the Cell and Machine Capacity Constraint is not Connsidered.

Objective function value = 87

Part	Operation	Assigned Machine
1	1	1
	2	2
	3	3
2	4	1
	5	2
3	6	2
	7	3
4	8	4
	9	5
	10	6
5	11	4
	12	5
6	13	4
	14	5

It can be noted that the part families can be different from the above and that too with the same objective function value. However, the number of exceptional elements may vary. For example, if the part families are (1,2} and {3,4,5,6} or {1,2,3,4}

and {5,6}, the minimum number of exceptional elements will be 3 and instead of 1 as for the grouping configuration determined earlier.

For the problem of simultaneous group determination when the constraints on permissible load on machines are not included and all the operations are to be performed within the cell, the problem will have the structure as of the model M5.2. For this case, the optimal operations assignment details are given in Table 5.8 and the configuration of the groups in Table 5.9.

Table 5.9: Group Configuration Details for the Case When the Operations are Carried Out Within the Cell and Machine Capacity Constraints is not Considered.

Group #	Part-Family	Machine-Cell
1	{1, 2, 3}	{1, 2, 3}
2	{4, 5, 6}	{4, 5, 6}

The operations assignment changes when machine capacity constraints are included as can be seen from the Table 5.10. However, the configuration of the groups remain the same. The present problem is equivalent to the model M5.3.

In case when the condition on perfect grouping is relaxed and machine capacity constraints are not included, both the operations assignment and grouping configuration changes. These are shown in Tables 5.11 and 5.12. This problem is best covered by the model M5.4. For the problem as characterized by the model M5.5 that relaxes the constraint on perfect grouping but considers

Table 5.10: Operation Assignment Details for the Case When the Operations are Carried Out Within the Cell and Machine Capacity Constraint is Included.

Objective function value = 85

Part	Operation	Assigned Machine
1	1	1
	2	2
	3	3
2	4	1
	5	2
3	6	1
	7	2
4	8	4
	9	5
	10	6
5	11	4
	12	5
6	13	4
	14	5

Table 5.15: Operation Assignment Details for the Case When Groups are Determined Minimizing Investment on Machines.

Objective function value = 69

Part	Operation	Assigned Machine
1	1	1
	2	2
	3	4
2	4	2
	5	2
3	6	1
	7	2
4	8	4
	9	5
	10	2
5	11	1
	12	5
6	13	4
	14	2

Table 5.16: Group Configuration Details for the Case When Groups are Determined Minimizing Investment on Machines.

Group #	Part-Family	Machine-Cell
1	{1, 4, 5, 6 }	{1, 2, 4, 5}
2	{2, 3}	{1, 2}

The same problem when solved using the heuristic GAO discussed in the Section 5.4.3.1 yields grouping as shown below. However, the constraints on machine capacity, group disjointedness and also on the size of the groups are not considered. The groups

are as follows:

Part Family # 1 = {1, 2, 3}

Machine Cell # 1 = {1, 2, 3}

Part Family # 2 = {4, 5, 6}

Machine Cell # 2 = {4, 5, 6}.

The operations assignment is, however, as shown in Table 5.17.

Table 5.17: Operation Assignment Obtained Using the Heuristic GAO.

Objective function value = 87

Part	Operation	Assigned Machine
1	1	1
	2	2
	3	3
2	4	1
	5	2
3	6	2
	7	3
4	8	4
	9	5
	10	6
5	11	4
	12	5
6	13	4
	14	5

It can be verified from the objective function values of the solutions given in the Tables 5.7, 5.8, 5.10, 5.11 and 5.12 that inclusion of the tighter constraints has negative effect on the objective function value.

5.7 SUMMARY AND CONCLUSION

In the present chapter, the generalized grouping problem is formulated as a graph partitioning problem. The mathematical models presented for the generalized grouping are shown to be better than the equivalent models presented in the Chapter III using p-median problem formulation approach as these require comparatively lesser number of constraints and variables. However, for the simple grouping problem, the graph partitioning based approach is shown not to be efficient.

It has been further shown that the use of similarity coefficients that find similarity between a pair of the same entities (such as parts, machines and process plans) is of much use in hierarchical grouping. However, it is of not great advantage in case of generalized grouping. Further, the graph partitioning based approach of hierarchical group formation is also of not much use.

The models presented are shown to be NP-complete. For some of these models, heuristics based on Lagrangian Relaxation are proposed. These models and heuristics consider the restriction on total number of groups to be formed.

The heuristics are also presented for determining the groups naturally without considering any restriction such as on number of groups, and size of part family and machine cell. The heuristic uses, of course, the basic approach of graph partitioning. The heuristic proposed is modified and extended for the simple grouping situation which besides determining the natural groups considers the restrictions on number of groups and the size of part families and machine cells. These heuristics can be suitably modified to include processing times of the operations and machine

capacity to yield grouping with operationally feasible solutions.

CHAPTER VI

LOADING PROBLEMS IN FMS

6.1 INTRODUCTION

Loading problem in flexible manufacturing systems (FMS), as in conventional manufacturing systems, involves decision making regarding the assignment of various resources to the operations of different parts that are to be produced during a planning period. In case of FMS, the problem, however, is comparatively more complex owing to (i) the versatility and flexibility of the FMS equipment such as computerized numerically controlled machines and machining centers, automated materials handling systems, robots, pallets, jigs and fixtures, etc., and (ii) the requirement of coordinated and integrated planning and control; and synchronized functioning of these equipment under the overall control of a central computer.

Loading and control policies have major impact on the performance of the manufacturing objectives. As observed by Stecke and Solberg (1981) from simulation of a real life FMS, the system performances depend heavily on the loading and the control strategies chosen to operate the FMS. Using the simulation model, they further showed that the results obtained are very much different from those of classical job shop scheduling studies. The loading problem in FMS, thus, involves simultaneous and integrated decision making related to various aspects such as part mix determination, production ratio determination, formation of part families and machine cells, and allocation of resources

(machines, tools, materials handling system, pallets, fixtures, etc.). The problem, therefore, is viewed as a part of system setup problem rather than being purely operational as is the case in conventional manufacturing systems (Ali S Kiran (1986), Stecke (1983)).

The processing equipment in FMS are capital intensive and thus non-productive times, such as, times taken for loading and unloading of parts, placement of tools into tool magazine from tool crib, tool magazine change over, etc. need to be minimized. Compared to these non-productive times, the time needed for picking and placing the tool from and to the tool magazine using automatic tool changer will, of course, be negligible. These non-productive times can, to a large extent, be minimized if the tool and operations are properly assigned to the machines.

Further, as mentioned earlier, the machines are capable of processing several types of operations. Hence, to a machine many operations requiring different types of tool can be assigned. These multifunctional machines have tool magazines or drums with slots to accommodate various types of tools of varying shapes and sizes. For this reason, the number of slots on the tool magazine often act as constraints for the number and type of operations (and thus for the parts) that can be assigned to a machine. Therefore, in FMS, due consideration needs to be imparted to the tool assignment keeping in view the requirements of the operations, the nature of the machines and the capacity of tool magazines. Real benefits of FMS can accrue only when the machine loading problem handles both the operations and the tool assignments decisions together in an integrated manner. Besides

finding the machine loading related decisions, specific recommendations also need to be made regarding fixtures and pallets to be used during machining and transportation of jobs. This would enable the central computer to identify and analyze the requirements of operations of various parts and thus ultimately would lead to an integrated implementation of real time scheduling.

The objectives of loading problem in FMS usually focus to satisfy several measures such as utilization of machines, throughput rate, and the total cost comprising machining, tooling and materials handling. The major objectives considered by various authors are as follows.

1. Minimization of the total cost of processing.
2. Maximization of throughput.
3. Balancing the workload on machines.
4. Balancing the workload per machine for a system of groups of pooled machines of equal sizes.
5. Unbalancing the workload per machine for a system of group of pooled machines of unequal sizes.
6. Minimizing the number of part movements from one machine to another (or, maximizing the number of consecutive operations on each machine).
7. Minimization of the tool movement.
8. Minimization of refixturing and other setup activities.
9. Filling the tool magazine as densely as possible.
10. Maximization of the sum of operations priorities.

The major constraints in loading include: the tool magazine capacity, the machine capacity, routing of parts, the number of available copies of a tool type and the allocation of fixtures and

pallets. In FMS, the considerations related to operations and tooling flexibility, and operations versatility of machines play important role in meeting the loading objectives and, of course, tend to make the problem complex.

A Brief Review of Previous Work

Loading problems with multiple objectives and multiple constraints have been attended by several researchers. Several types of optimization formulations such as 0-1 integer programming (pure, mixed and nonlinear) have been proposed. A wide range of methodology including mathematical programming techniques, branch and bound, Lagrangian relaxation have been suggested to obtain solution of the loading problem. In addition, in response to analytical complexity of the problem, heuristic procedure and simulation appear to dominate the list of solution methodologies.

As mentioned earlier, solutions to the loading problems in FMS need to be obtained in conjunction with certain other operational and setup decisions for an effective integration and coordination of various elements of the system. Several researchers have presented the integration of loading problem with scheduling, grouping, part mix determination, etc.

In the subsequent paragraphs, a brief review of some of the earlier work is presented. The pure loading problem is taken first to be followed by its integration with other functions, and then the structural approaches.

Stecke (1983) has formulated the loading problem as a 0-1 nonlinear mixed integer programming problem. For the objectives 3, 4, 5 and 6 listed earlier, operations are assumed to have fixed routing, while for the objectives 9 and 10, the condition on

capacities of the machines and the tool magazines.

A branch and bound approach was suggested by Stecke and Berrada (1986) for the minimization of the workload unbalance on machines. The problem considered is the same as one by Stecke (1983), but the approach for solving the problem is different. This problem and the solution methodology was extended by Shanker and Srinivaslu (1986) for the case of random FMS. For the problems where each part had only one operation, the objective is to minimize the workload unbalance and job lateness, whereas for the problems where parts had multiple operations, the objective is the maximization of the assigned workload. They also suggested three heuristics each for a different problem situation having bi-criteria objective of minimizing the workload unbalance and maximizing the throughput. In the first heuristic, primary objective is to minimize the unbalance while in the other two, maximization of the throughput is considered. For the first two heuristics critical resource is tool magazine capacity, whereas for the third it is the machine capacity.

Loading problems were also considered by numerous other researchers. Na et al (1987) considered the problem of minimizing tool transfer between versatile machines where a part is to be completely processed on a single machine only. The constraints were imposed on the number of copies available of each tool type, tolerable workload unbalance and tool magazine capacity. It is assumed that each tool type requires the same amount of tool slots, and the formulation presented considers the requirement of a single slot by each tool type.

O'grady et al (1987) considered the problem of choosing

compatible subset of candidate orders for processing in a FMS subject to the machine capacity available in the planning period. Parts have unique job routing. The slot requirement of each tool is unity. A structural framework is also suggested for resolving potentially conflicting multiple performance objectives.

Sarin and Chen (1987) formulated the problem as 0-1 integer programming problem of minimizing cost of processing considering constraints on the capacities of machines and tool magazines, and on the life of cutting tools. The salient difference from the model of Kusiak (1985) is an additional restriction imposed on the number of different operations that can be assigned to a particular cutting tool. The formulation assumed fixed routing for each of the parts. Lagrangian relaxation of the problem was proposed and the solution is obtained using subgradient algorithm. The distinguishing characteristic of the formulation is the consideration given to the tool and machine combination where a tool can be assigned to only a limited number of machines and the jobs which require certain tools can be performed on only those machines which support these tools. The tools of the same type are given a different number and are considered separately making the assignment of duplicate tools to the same machine possible and leading to increased tooling flexibility. It was claimed that the assignment of duplicate tools will allow the system to have the potential of routing parts alternatively through the machines for real time operational control. The formulation, however, does not seem to explicitly take care of such possibility of alternate routing.

Chakrabarty and Shtub (1987) formulated the problem as a

nonlinear integer programming problem where the sum of the costs on setup and inventory carrying was to be minimized subject to the constraints on pallet allocation, tool magazine capacity and on the completion of all the operations within the planning horizon.

Lashkari et al (1987) considered a somewhat different configuration of FMS where if a part for the processing of its next operation requires a different machine and/or a different fixture, then the part is brought to the central storage and from here it is transported to the desired machine. The objectives considered are the minimization of: (i) traffic and (ii) refixturing activities. The mathematical expressions representing the allocation of certain group of operations of the same part on a machine and also the tool slot saving are nonlinear in nature. The problem with the constraints on the number of copies available of each tool type and unique routing is solved after linearizing the nonlinear terms. The consideration for the tool slot consumption is the same as used by Stecke (1983). Wilson (1989) proposed a reformulation of this problem which avoided the nonlinearities of the formulation proposed by Lashkari et al (1987). However, the slot requirement of an operation is conside-red to depend on the machine on which it is to be performed.

Integrated loading and grouping problems were considered in a MRP II framework by Mazzola et al (1987) including constraints on machine capacity and the tool magazine capacity with slot saving.

An effort was made by Greene et al (1986) for integrating loading problem with scheduling considering the operations sequence. Each part is to be assigned to one of the cells which operate independently. Loading-cum-scheduling decision problem is

mathematically modeled. The objectives considered are the minimization of: (i) makespan time, (ii) mean flow time, and (iii) mean lateness.

A hierarchical approach for solving the problem of grouping and loading was used by Stecke (1986). At the first stage, aggregated level of details are used and certain results are obtained using closed queuing network model which are then used by the lower level in hierarchy that considers all the details of the problem.

Kiran (1986) considered the operation assignment problem along with part mix selection problem for the objective of maximizing throughput. The constraint considered are on the machine and the tool magazine capacities, and also on the number of copies available of each tool type. In addition, a constraint is imposed on the number of machine changes for each of the parts. The way in which fixture allocation constraint is imposed, it does not affect either the selection of a part for production or the assignment of an operation to a machine except when a limit on the total number of fixture changes is considered. The solution to the problem is determined by decomposing the relaxed problem into four sub-problems which are solved easily.

Comments on Loading Considerations

From the descriptions presented in the previous paragraphs, it can be observed that majority of the researchers have considered the loading problems with the constraints on tool magazine capacity and processing time availability on machines. Only few have included constraints on pallet and fixture allocation.

The constraint related to the tool magazine capacity makes the formulation quite complex because of the nonlinear terms (Stecke (1983)). The net slot requirement of the tools on a machine is computed as the sum of slots required by the individual tools to be used for the various operations assigned to the machine minus the slots saved. The saving in slots requirement may arise due to (i) tooling flexibility when more than one operation may require some of the same tools and thus only minimal number of tools are assigned (tool duplication), and (ii) physical placement of tools on the magazine that may result into some efficient knapsacking of inter-slot space (tool overlapping). Such saving arises because of the varying flank size of tool shanks that are to be placed on the magazine in the equispaced slots.

There are several difficulties involved in the computation of tool slot requirement. First of all, computation of slot savings will involve consideration of various permutations of arrangement of tools in the tool magazine and thus complex expressions involving nonlinear terms. On the other hand, if the machines have different inter-slot spacings on their magazines, the problem of computing slot requirement would become further complex. Hence, in the absence of a specific method for counting the precise slot requirements of tools, it may become difficult to find tool slot savings for a group of tools when these are placed together on a magazine. It should further be noted that in the nonlinear expression showing the total tool slot requirement, a term itself represents the tool slot saving for the tools that are put together in the same tool magazine. Since the difference

between the sum of the individual slot requirement and the maximum saving in the total tool slot requirement obtained by the placing the tools properly into the tool magazine gives the net slot requirement of the tools, other intermediate combinations of tools and their resulted saving need not be considered. Since the tools that will be assigned to a machine are not known apriori and thus this kind of simplification would not be feasible.

Further, since the number of different types of operations that can be performed on a machine depends upon the number of different types of tool that are placed in the tool magazine of limited slots; in order to reap the maximum benefit tools should be placed into tool slots judiciously, and for that the slot requirement of each tool type must be correctly expressed and used in the formulation. In a later section of this chapter, efforts have been made to suggest few methodologies for counting the slot requirements of the tools.

The problem of allocation of pallets and fixtures has been viewed to depend on the operation of a part, paying no explicit consideration to the machines on which the operation is carried out. Pallets and fixtures can be universal or part-specific and they can take up more than one part at a time. In case of part-specific fixtures and pallets, the allocation decision is trivial. The universal pallets and fixtures can be treated as general resources and incorporation of such consideration will make the problem of loading further complex to analyze and solve. The problem of assignment of pallets and fixtures can be taken up at scheduling stage where considering the nature, type and use of the pallets and fixtures, one can find these decisions with lesser

efforts.

Scope and Organization of the Present Chapter

In the present chapter, attempts have been made to formulate the problem of machine loading and tool allocation for two situations. One in which an operation is restricted to be processed using only a single combination of a tool and a machine (unique routing), and the other in which more than one such combinations are included in the final assignment (multiple routing). For the first case of unique job routing the problem of scheduling may get simplified to some extent, but throughput rate may go down due to starvation and blocking at machines. In the case of multiple routing, the jobs will obviously have, more routing flexibility, and thus the production system will have comparatively lesser congestion and better throughput rate.

The models developed consider constraints on the capacities of the tool magazines and the limit on the number of copies available of each tool type. The capacity of the machines are also considered in the problem formulation. Besides the objectives of minimizing the processing cost and overload on the machines, and maximizing the processing flexibilities, the objective of minimizing the maximum processing workload on the machines is also considered. This objective while minimizes indirectly the unbalance in the workloads of the machines, it also provides an estimate about the minimum time required to process all the operations on all parts.

The processing flexibility is provided by allowing assignment of operations to more than one machine where each machine itself is sufficient to process that operation on all the units of the

performance for the problem of minimizing the system unbalance using the corresponding heuristic and sequential loading method with different dispatching rules.

Kusiak (1985) considered the tool magazine flexibility in which to a machine more than one tool magazines are allocated and depending upon the requirements of the operations, magazines are changed over. However, no explicit considerations are given for tool slot saving resulting from tool duplication and overlap which was one of the major considerations accorded by Stecke (1983). The objective is to minimize the sum of cost of processing all the operations and the total penalty cost of tool magazine changeovers over various machines. In the other paper (Kusiak (1985)), he proposed four more models for tool allocation and machine loading in FMS. All the models consider the objective of minimizing the cost of processing the operations over all the batches of parts. All the operations in a batch were apparently of the same type and were allowed to be performed on more than one machine. In the first model, a constraint is imposed on the total processing time which can be imparted by a machine to a batch, whereas in the second model a constraint is imposed on the tool magazine capacity, and in the third model constraints are imposed to consider machine capacity and tool life. The fourth model has the features of all the models. The solution to these models are obtained using subgradient algorithm.

Later, while describing a structural approach for solving FMS problems, Kusiak (1985) proposed two more loading models, one for unique job routing and the other for alternate routing. The objective is the minimization of cost with the constraints on the

machines which (i) may be versatile or (ii) may not necessarily be versatile. The details of the problems, the corresponding formulations and examples are given for each of these two different situations in the subsequent subsections.

6.3.1 All Machines Versatile

In the case where in an FMS, all the machines are versatile, following additional assumptions are made.

(i) An operation can be performed on any machine.

(ii) A tool can be loaded on any machine.

(iii) The machines though functionally versatile need not be identical in terms of their size, power and machining characteristic (speed, feed, etc.).

In the following subsections, three models are presented each considering a different objective for the loading problem.

6.3.1.1 MINIMIZATION OF PROCESSING COST

OBJECTIVE FUNCTION

The objective of minimization of the total processing cost can be written as :

$$\text{Minimize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{j=1}^{M} \sum_{k \in t(i)} c_{nijk}\, x_{nijk}. \tag{6.1}$$

CONSTRAINTS

The constraints on the loading problem are described below.

(i) <u>Load on machines</u>

The processing load of the operations assigned to any machine should not exceed the available capacity. This constraint is expressed as:

$$\sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{k \in t(i)} T_{nijk} \, x_{nijk} \le C_j \quad j = 1, \ldots, M. \tag{6.2}$$

(ii) Unique routing and tooling

An operation of a part in the final loading decision is carried out on only one machine and using only one tool type. This constraint is written as:

$$\sum_{j=1}^{M} \sum_{k \in t(i)} x_{nijk} = 1 \quad n = 1, \ldots, N \text{ and } \forall \, i \in B(n). \tag{6.3}$$

(iii) Slot used by the tools allocated to a machine

As mentioned in the previous section, the tool slot constraint contributes to the maximum complexity in formulation of the loading problem when savings in slot requirement due to tool duplication and overlapping are incorporated (Stecke (1983)). The saving due to tool duplication need not be considered if it is ensured that to a tool magazine only one copy of a tool type is assigned. The consideration of slot saving due to tool overlap will largely depend upon the method for counting the slot requirement of each tool type as can be seen from the following discussions.

In Figure 6.1, a tool magazine is shown with five tools mounted on it. The small circles represent tool shank periphery. If slot requirement of a tool is taken on the basis of slots physically covered by them, then the slot requirements of the tools I, II, III, IV and V will be 1, 1, 3, 1 and 1, respectively. For the present arrangement of the tools into the tool magazine, the slot requirement determined for each tool type is alright. However, the problem arises when some different physical

arrangement of the tools in the magazine is considered. For example, when tool II is placed between any pair of the tools I, III and V, then it requires only one slot. But when it is placed in between tools IV and V, it requires more than one tool slot.

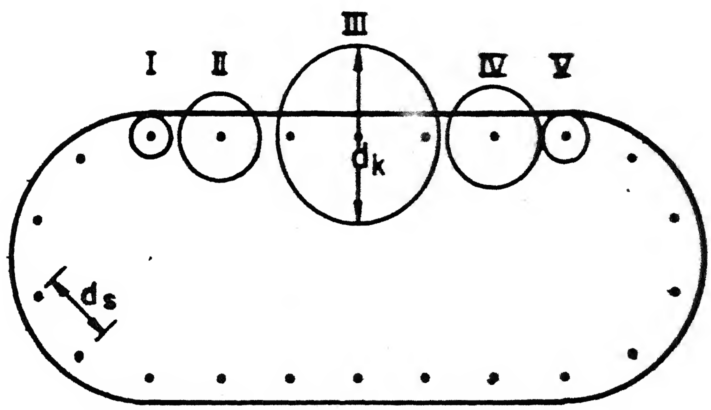


Figure 6.1: Tool Drum with Tools Mounted on it.

The difficulty observed in the determination of tool slot requirement can be overcomed using the following two approaches. While describing the two approaches, the rationale behind each of these is also discussed.

(a) The slot requirement of a tool type is given as:

$$s_k = 2\left\lceil \frac{d_k}{2d_s} \right\rceil + 3 \qquad k = 1, \ldots, T.$$

where d_k and d_s are as shown in Figure 6.1 and $\lceil x \rceil$ represents a largest integer smaller than or equal to x. It can be noted that the above expression tries to find the slot requirement of a tool as the minimum number of slots within which the tool can be accommodated. For example, the slot requirement of the tools I, II, III, IV and V shown in the Figure 6.1 will be 3, 3, 5, 3 and 3 respectively. It can also be seen that the slot requirements of tools determined using the above expression will always be odd.

For the two tools that are placed together at $\underline{k}1$-th and $(\underline{k}1+1)$-th position in the magazine, the slot saved $S_{\underline{k}1,\underline{k}1+1}$ can be expressed as:

$$S_{\underline{k}1,\underline{k}1+1} = s_{\underline{k}1} + s_{\underline{k}1+1} - 2 - \left\{\begin{array}{l}\text{Number of slots whose centers} \\ \text{have been covered by the two} \\ \text{tools when placed quite close} \\ \text{to each other}\end{array}\right\}$$

From the above definition of the slot saving, it can be seen that as expected the actual slot requirement will depend upon the arrangement of the tools in the magazine. If the adjacent tools are properly placed, then each such pair of tools that are placed close together will result into a saving of two slots at the most. In any case, the two tools placed close together will result into slot saving of one. As shown in Figure 6.2, the slot requirements of the tools I, II, III and IV are 5, 3, 5, and 3, respectively. The tools I and II are not compatible in the sense that their shank

peripheries are interfering, and thus result into slot saving of one ; whereas compatible tools I and III placed closely yield slot saving of two.

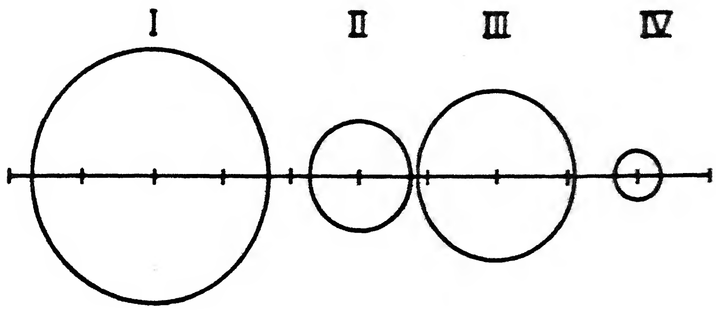


Figure 6.2: An Example for Illustrating Tool Slot Saving.

From the above, it can be concluded that the slot requirements of the tools determined using the scheme discussed in the previous paragraphs are always odd, and the number of slots saved for a pair of tools placed closed together is always positive.

The scheme mentioned here is somewhat different from that of Stecke (1933), and of Stecke and Berrada (1986) where slot requirements of tools are not necessarily represented by odd numbers, and savings for various pairs of tools are also not necessarily positive.

The total slot requirement for a set of tools K, i.e. s_K can be determined using the expression given below. It is assumed that the set K contains all the tools that are to be placed into a tool magazine. Further, the tools are placed in the magazine in a order in which they appear in the set K. In the following expression, $s_{\underline{k}}$ denotes slot requirement of the tool at $\underline{k}$-th position.

$$s_K = \sum_{\underline{k}=1}^{|K|} s_{\underline{k}} - \left\{ \sum_{\underline{k}=1}^{|K|-1} s_{\underline{k},\underline{k}+1} + s_{|K|,1} \right\}$$

It can be observed from the above expression that the total slot requirement for a set of tools is equal to the sum of individual slot requirement of tools less the savings for each pair of adjacent tools. Hence, the total slots consumed will be equal to the number of slots whose centers have been covered by the tools. The minimum total slot requirement can be different from that obtained using the above expression and can be determined by solving the related problem as a symmetric travelling salesman problem as shown by Agrawal and Shanker (1988).

It should be noted that the slot requirement of a tool may vary with the tool magazine, and so would the total slot requirement for a set of tools.

(b) The above mentioned scheme for counting the total slot requirements of tools involves determination of slot savings and arrangement of tools in the tool magazine. The formulation incorporating such saving will be complex making the loading problem more difficult to solve. The amount of slot saving between a pair of tools depends upon their shank diameters. If the two tools are compatible, then the saving is of two slots, else of one showing that the number of slots saved depend upon the shank diameters of the tools causing or not causing interference. In case of compatibility (no interference), no slot except that covered by the tools of a pair will be unused; whereas in case when the two tools interfere, one slot will be forced to remain idle besides those which are covered by the tools themselves.

In general, the shank diameters of the tools may be different from each other. If the variation in the compatibility between two tools varies uniformly, then the saving in the slot requirement of the two tools need not be considered provided slot requirements of the tools are determined using the following relation:

$$s_k = \left\lceil \frac{d_k}{d_s} \right\rceil + 1 \qquad k = 1, \ldots, T.$$

If the slot requirements are computed using the relation given above, then for the tools I, II, III and IV (shown in the Figure 6.2) the slot requirements will be 4, 2, 3 and 1 respectively. It can be noticed that the number of slots whose centers have been physically covered by the tools I and II are different from their respective computed slot

(b) The above mentioned scheme for counting the total slot requirements of tools involves determination of slot savings and arrangement of tools in the tool magazine. The formulation incorporating such saving will be complex making the loading problem more difficult to solve. The amount of slot saving between a pair of tools depends upon their shank diameters. If the two tools are compatible, then the saving is of two slots, else of one showing that the number of slots saved depend upon the shank diameters of the tools causing or not causing interference. In case of compatibility (no interference), no slot except that covered by the tools of a pair will be unused; whereas in case when the two tools interfere, one slot will be forced to remain idle besides those which are covered by the tools themselves.

In general, the shank diameters of the tools may be different from each other. If the variation in the compatibility between two tools varies uniformly, then the saving in the slot requirement of the two tools need not be considered provided slot requirements of the tools are determined using the following relation:

$$s_k = \left\lceil \frac{d_k}{d_s} \right\rceil + 1 \qquad k = 1, \ldots, T.$$

If the slot requirements are computed using the relation given above, then for the tools I, II, III and IV (shown in the Figure 6.2) the slot requirements will be 4, 2, 3 and 1 respectively. It can be noticed that the number of slots whose centers have been physically covered by the tools I and II are different from their respective computed slot

The constraints (6.6) and (6.7) are joined together and are represented by the following single constraint.

$$Z_{jk} - x_{nijk} \geq 0 \qquad i = 1, \ldots, N;\ \forall\ i \in B(n);\quad j = 1, \ldots, M;\ \forall\ k \in t(i). \qquad (6.8)$$

DECISION VARIABLES

$$x_{nijk} = 0 \text{ or } 1 \qquad n = 1, \ldots, N;\ \forall\ i \in B(n)\quad j = 1, \ldots, M;\ \forall\ k \in t(i) \qquad (6.9)$$

$$Z_{jk} = 0 \text{ or } 1 \qquad j = 1, \ldots, M;\ k = 1, \ldots, T. \qquad (6.10)$$

MODEL (M6.1)

The formulation of the problem is summarized below.

$$\text{Minimize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{j=1}^{M} \sum_{k \in t(i)} c_{nijk}\, x_{nijk}$$

Subject to: (6.2), (6.3), (6.4), (6.5), (6.8), (6.9) and (6.10).

The number of variables and constraints used in the above formulation are as follows.

$$\text{Number of variables} = M \left\{ \sum_{n=1}^{N} \sum_{i \in B(n)} |t(i)| \right\}$$

$$\text{Number of constraints} = M \left\{ \sum_{n=1}^{N} \sum_{i \in B(n)} |t(i)| \right\} + \sum_{n=1} |B(n)| + 2M + T$$

EXAMPLE

For illustration, an example (Example 6.1) with the data given in Tables 6.1, 6.2 and 6.3 is solved on IBM compatible PC/AT using LINDO software. The results on operations assignments are

given in Table 6.4, and that on tools allocations and capacity utilization in Table 6.5.

6.3.1.2 MINIMIZATION OF MAXIMUM LOAD ON MACHINES

OBJECTIVE FUNCTION

The objective function can be expressed as :

$$\text{Minimize} \left[\underset{j}{\text{maximum}} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{k \in t(i)} T_{nijk}\, x_{nijk} \right].$$

The above objective can be equivalently written in the following form.

$$\text{Minimize } L, \tag{6.11}$$

where

$$\sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{k \in t(i)} t_{nijk}\, x_{nijk} \le L, \quad j = 1, \ldots, M. \tag{6.12}$$

CONSTRAINTS

For the objective of minimizing the maximum workload on machines, the constraint on the load on machines bears no relevance, and thus is not included. The other constraints on unique routing and tooling, slots used by the tools allocated to a machine, copies of each tool type and on tool allocation are the same as described in the Section 6.3.1.1.

MODEL (M6.2)

The formulation of the loading problem for the objective of minimizing the maximum workload is briefly described below.

Minimize L

Subject to: (6.12), (6.3), (6.4), (6.5), (6.8), (6.9) and (6.10).

Table 6.1: Resource Requirements and Related Costs for the Operations of the Parts (Example 6.1).

N = 5; M = 3 and T = 5

Part (n)	Operation ($i \in B(n)$)	Tool type ($k \in t(i)$)	Processing cost on machine (C_{nijk}) 1	2	3	Processing time on machine (T_{nijk}) 1	2	3
1	1	1	4	6	10	6	10	4
		2	5	12	3	7	3	8
	2	3	6	7	5	7	5	6
	3	4	10	11	4	11	4	10
2	4	2	7	6	3	8	5	8
		2	12	5	4	3	2	8
		5	1	2	10	13	11	3
	5	4	9	6	4	7	8	3
3	6	2	2	3	5	8	7	1
		3	11	4	9	3	3	7
	7	5	5	7	11	2	9	5
		1	2	3	6	10	7	5
	8	2	4	5	1	11	3	9
		4	2	3	4	10	11	8
	9	1	5	6	9	2	3	7
		2	11	3	7	3	6	5
4	10	5	2	1	3	5	7	9
		1	11	4	8	3	7	10
	11	3	9	4	9	3	8	5
5	12	2	12	4	6	3	7	6
	13	2	2	1	6	4	5	9
		4	7	9	4	8	4	2
	14	1	10	8	7	7	9	9

Table 6.2: Available Copies and Slot Requirements of Various Tool Types (Example 6.1).

Tool Type (k)	Slot Requirement (s_k)	Number of Copies (V_k)
1	3	2
2	2	3
3	3	3
4	1	2
5	2	1

Table 6.3 : Machine and Tool Magazine Capacity (Example 6.1).

Machine (j)	Available Processing Time (C_j)	Tool Magazine Capacity (U_j)
1	35	5
2	35	5
3	50	5

Table 6.4: Operations and Tools Assignments for the Objective of Minimizing Total Processing Cost (Example 6.1).

Part	Operation	Machine	Tool Type
1	1	3	2
	2	2	3
	3	3	4
2	4	3	2
	5	3	4
3	6	1	2
	7	1	1
	8	3	2
	9	2	2
4	10	3	5
	11	2	3
5	12	2	2
	13	2	2
	14	1	1

The total optimal processing cost = 51 units

Table 6.5: Tool Allocation and Capacity Utilization for the Objective of Minimizing the Total Processing Cost (Example 6.1).

Machine	Types of Tool Allocated	Capacity Used	Slots Consumed
1	1,2	25	5
2	2,3	33	5
3	2,4,5	47	5

The number of decision variables and constraints involved in the above formulation are the same as that in the model M6.1.

Table 6.6: Operation Assignment Detail for the Objective of Minimizing the Maximum Workload on Machines (Example 6.1).

Part	Operation	Machine	Tool Type
1	1	1	1
	2	2	3
	3	3	4
2	4	2	3
	5	3	4
3	6	2	3
	7	1	5
	8	2	2
	9	1	1
4	10	1	1
	11	3	3
5	12	2	2
	13	3	4
	14	1	1

The maximum workload = 20 units

EXAMPLE

The numerical example given in the Section 6.3.1.1 (Example 6.1) solved for the objective of minimizing the maximum workload yields assignment of operations as given in Table 6.6. The tool allocation and capacity utilization is shown in Table 6.7. The maximum workload on the machines for this assignment in 20 units whereas for the objective of maximizing the total processing cost is as high as 47 units.

Table 6.7: Tool Allocation and Capacity Utilization for the Objective of Minimizing the Maximum Workload on Machines (Example 6.1).

Machine	Types of Tool allocated	Capacity Used	Slots Consumed
1	1,5	20	5
2	2,3	20	5
3	3,4	20	4

6.3.1.3 MINIMIZATION OF THE SUM OF OVERLOADS ON MACHINES

In certain manufacturing situations, the time allocated initially on some of the machines may not be sufficient to process all the operations on all the units of parts even when the operations can be performed on more than one machine because of the multifunctionality of the machines. Therefore, for the complete processing some of the machines would require to be overloaded. It is natural to desire that the total weighted workload on the machines be as minimum as possible. The objective of minimizing the the sum of overload on machines can be written

as:

$$\text{Minimize} \quad \sum_{j=1}^{M} W_j O_j.$$

Except the constraint on the load on the machines, the other constraints will be the same as given in the Section 6.3.1.1. The constraint on the load of the machine will be:

$$\sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{k \in t(i)} T_{nijk} X_{nijk} - O_j \le C_j \qquad j = 1, \ldots, M$$

where $$O_j \ge 0 \qquad j = 1, \ldots, M.$$

For the example given in the Section 6.3.1.1, a solution has already been obtained which does not require additional time on any of the machines. The solution will remain optimal even for the objective of minimizing the sum of overload on machines. However, a different solution may be obtained when the sum of overload and underload on machines is to be minimized.

6.3.2 Machines not Necessarily Versatile

The manufacturing system considered in the Section 6.3.1 is assumed to have the machines that are capable of processing all the operations. In practice, all the machines in a FMS need not be versatile to this extent and may process only limited number of operations. A production system of this kind is described by the following assumptions in addition to those described in the beginning of the Section 6.3.

(i) An operation can be performed only on limited machines.

(ii) A tool type can be mounted only on some machines because of the limitations on the shape, size and functional characteristics of the tool.

(iii) The tool needed to perform an operation may depend upon the machine to be used for the processing.

Notations

Following are the notations in addition to that defined in the Section 6.2 used for formulating the loading problem.

$\underline{B}(j)$ = set of the indices of the operations that can be performed on machine j

$\underline{t}(j)$ = set of tool types that can be allocated to machine j

m(i) = set of machines that are capable of performing operation i

$\underline{m}(k)$ = set of machines on which tool type k can be loaded.

In the following subsections, models are developed each for a different loading objective.

6.3.2.1 MINIMIZATION OF PROCESSING COST

OBJECTIVE FUNCTION

The objective of minimization of the total processing cost is expressed as:

$$\text{Minimize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{j \in m(i)} \sum_{k \in t(i) \cap \underline{t}(j)} c_{nijk}\, x_{nijk}. \qquad (6.13)$$

CONSTRAINTS

The expressions for the various constraints are as follows :

(i) Load on machines

For the case where machines may be not be versatile, and all the tools may not be loaded on all the machines, the constraint that total workload of the operations assigned to a machine should

Table 6.8: Resource Requirements and Related Costs for the Operations on the Parts (Example 6.2).

$N = 5$, $M = 5$ and $T = 5$

Part n	Operation $i \in B(n)$	Capable Machines $j \in m(i)$	Capable Tool Types $k \in t(i)$	Feasible Combinations of Machines and Tool Types and Corresponding Processing Processing Costs and Times $(j, k, C_{nijk}, T_{nijk})$
1	1	1,3,5	1,2	(1,1,7,8),(3,2,6,5),(5,1,3,8)
	2	1,4	2,3	(4,2,12,3)
	3	1,2	3,4	(1,4,6,8),(2,3,4,3)
2	4	2,5	3	(2,3,9,7),(5,3,2,8)
	5	2,3	3,4,5	(2,3,11,3),(2,5,4,3),(3,5,7,9)
3	6	4,5	1,3	(5,1,2,10),(5,3,9,4)
	7	3,4	2,4,5	(3,2,3,7),(3,5,4,8),(4,2,6,5),(4,4,3,7)
	8	2,3	2,4	(3,2,6,5)
	9	1,3	3,5	(3,5,4,11)
4	10	4,5	3	(5,3,5,3)
	11	2	1,5	(2,1,1,9),(2,5,2,4)
5	12	2	3,5	(2,3,2,10),(2,5,1,7)
	13	3	1,2,5	(3,2,3,11),(3,5,9,11)
	14	1,3	2,4	(1,4,4,8),(3,2,2,7)

Table 6.9: Available Copies and Slot Requirements of Various Tool Types (Example 6.2).

Took Type (k)	Slot Requirement (s_k)	Number of Copies (V_k)	Compatible Machines Set (m(k))
1	3	2	1,2,5
2	2	3	3,4
3	3	3	2,5
4	1	2	1,4
5	2	2	2,3

Table 6.10: Machine and Tool Magazine Capacity, and Compatibility Details (Example 6.2).

Machine (j)	Available Processing Time (C_j)	Tool Magazine Capacity (U_j)	Compatible Set of Tool Types ($\underline{t}(j)$)	Compatible Set of Operations ($\underline{B}(j)$)
1	20	5	1,4	1,2,3,9,14
2	15	5	1,3,5	3,4,5,8,11,12
3	35	5	2,5	1,5,7,8,9,13,14
4	15	5	2,4	2,6,7,10
5	20	5	1,3	1,4,6,10

Table 6.11: Operation Assignment Details for the Objective of Minimizing the Total Processing Cost (Example 6.2).

Part	Operation	Machine	Tool Types
1	1	1	1
	2	4	2
	3	1	4
2	4	5	3
	5	2	5
3	6	5	3
	7	4	4
	8	3	2
	9	3	2
4	10	5	3
	11	2	5
5	12	2	5
	13	3	2
	14	3	2

The optimal processing cost = 66 units

Table 6.12: Tool Allocation and Capacity Utilization for the Objective of Minimizing the Total Processing Cost (Example 6.2).

Machine	Allocated Tool Types	Capacity Used	Slots Consumed
1	1,4	16	4
2	1,5	14	5
3	2,5	34	4
4	2,4	10	3
5	3	15	3

6.3.2.2 MINIMIZATION OF MAXIMUM WORKLOAD ON MACHINES

OBJECTIVE FUNCTION

The objective of minimizing the maximum workload on machines is written in the following form.

Minimize L

such that

$$\sum_{n=1} \sum_{i \in B(n) \cap \underline{B}(j)} \sum_{k \in t(i) \cap \underline{t}(j)} T_{nijk}\, x_{nijk} \le L \quad j=1, \ldots, M \qquad (6.21)$$

CONSTRAINTS

In this case also, the consideration of the constraint on the workload of machines has no relevance. The other constraints are the same as described in the Section 6.3.2.1.

MODEL (M6.4)

The formulation of the problem will have the objective function and the constraints as described below.

Minimize L

Subject to: (6.21), (6.15), (6.16), (6.17), (6.18), (6.19) and (6.20).

EXAMPLE

For the objective of minimizing the maximum workload on machines, the solution to the example (Example 6.2) described in the Section 6.3.2.1 is given in Tables 6.13 and 6.14.

It can be noticed that for the objective of minimizing the processing cost, the maximum workload is 34 units corresponding to machine 3 (see Table 6.12), whereas as shown in the Table 6.14, the maximum workload for the objective of minimizing the maximum workload is 27 units corresponding to machine 3.

6.3.2.3 MINIMIZATION OF THE SUM OF OVERLOADS ON MACHINES

For the case when all the machines are not necessarily versatile, the formulation of the problem for the objective of minimizing the total weighted overload on machines is as follows.

Table 6.13: Operation Assignment Details for the Objective of Minimizing the Maximum Workload on Machines (Example 6.2).

Part	Operation	Machine	Tool Type
1	1	1	1
	2	4	2
	3	2	3
2	4	2	3
	5	2	5
3	6	5	3
	7	4	2
	8	3	2
	9	3	5
4	10	5	3
	11	2	5
5	12	2	5
	13	3	2
	14	1	4

The maximum workload = 27 units

Table 6.14: Tool Allocation and Capacity Utilization Detail for the Objective of Minimizing the Maximum Workload on Machines (Example 6.2).

Machine	Allocated Tool Types	Capacity Used	Slots Consumed
1	1,4	16	4
2	3,5	24	5
3	2,5	27	4
4	2	8	2
5	3	7	3

$$\text{Minimize} \quad \sum_{j=1}^{M} W_j O_j .$$

Subject to:

$$\sum_{n=1}^{N} \sum_{i \in B(n) \cap \underline{B}(j)} \sum_{k \in t(i) \cap \underline{t}(j)} t_{nijk}\, x_{nijk} - O_j \le C_j \qquad j = 1, \ldots, M$$

and (6.15), (6.16), (6.17), (6.18), (6.19), (6.20),

and $\quad O_j \ge 0 \qquad\qquad j = 1, \ldots, M.$

The solution given in the Section 6.3.2.1 for the Example 6.2 is also feasible for the present case. Since the objective function value for this solution is minimum (equal to zero), the solution will be optimal even for the present problem.

6.4 LOADING MODELS WITH ALTERNATE ROUTING

The loading models developed in the previous section provide solutions for those problem situations where all the operations on all the units of a part that require the same machining parameters and setup support are to be processed using only one combination of a tool and a machine. The loading models that will be developed in this section, this restriction is removed and multiple routing for the parts is allowed. The various occurrences of an operation whether on the same unit of a part or on the different units of the part are considered separately for assignment, and thus may get assigned to different machines and may require different tool types. The alternative routes for a part can be obtained in a number of ways as follows.

(i) For no operation duplicate assignment is allowed and for all the operations of all the parts exactly required number of assignments are made. For example, for operation i of part

n the total number of assignments to be made will be equal to $(a_n.b_{ni})$.

(ii) Duplicate assignments of operations are allowed. That is, an operation of a part can be assigned to more than one machine.

In the following two subsections, formulations of loading problems incorporating the alternate routings as mentioned above are presented.

6.4.1 Operations not Duplicated

In this case though the condition on unique job routing and tooling is relaxed, but the extra assignment of operations are prevented. Flexibility is provided by assigning various occurrences of an operation of a part to more than one machine and thus by allowing alternative routes for the operations of the various parts. Assignments of the operations to machines in this way not only brings in flexibility, but also helps in reducing the processing cost and unbalance in the workloads of the machines, and also in achieving a better machine utilization. These advantages are obtained by utilizing the remaining capacities of the machines for processing on a portion of certain batches, the operations for which these machines are definitely a better choice.

The objective function and the constraints for the objective of minimizing the processing cost are as given below.

MODEL (M6.5)

$$\text{Minimize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{j \in m(i)} \sum_{k \in t(i) \cap \underline{t}(j)} c_{nijk}\, x_{nijk} \qquad (6.22)$$

Subject to:

$$\sum_{n=1}^{N} \sum_{i \in B(n) \cap \underline{B}(j)} \sum_{j \in m(i) \cap \underline{t}(j)) \cap \underline{t}(j)} t_{nijk} X_{nijk} \leq C_j$$

$$j=1, \ldots, M \qquad (6.23)$$

$$\sum_{j \in M(i)} \sum_{k \in t(i) \cap \underline{t}(j)} X_{nijk} = a_n b_{ni} \qquad n = 1, \ldots, N; \; \forall i \in B(n) \qquad (6.24)$$

$$a_n . b_{ni} Z_{jk} - X_{nijk} \geq 0 \qquad n = 1, \ldots, N; \; \forall i \in B(n); \; \forall j \in m(i); \; \forall k \in t(i) \cap \underline{t}(j) \qquad (6.25)$$

$$X_{nijk} \geq 0 \text{ and integer} \qquad n = 1, \ldots, N; \; \forall i \in B(n); \; \forall j \in m(i); \; \forall k \in t(i) \cap \underline{t}(j) \qquad (6.26)$$

and (6.16), (6.17) and (6.20).

The constraint (6.23) is related with machine capacity, whereas the constraint (6.24) with the completion of all the operations of the parts. The constraint (6.25) ensures that the required tools are assigned on the machines and also serves the purpose of linking the two variables X_{nijk} (general integer variable) and Z_{jk} (binary integer variable).

In the above model, there is a mix of binary and general integer variables. The formulation will have only binary variables provided X_{nijk}'s are replaced by sum of binary variables each for a single operation. Such representation will usually involve large number of binary variables (i.e. $a_n . b_{ni}$ number of variables for operation i of part n), and thus may not be desirable. However, using the scheme given below, variable X_{nijk} can be expressed using much lesser number of binary variables.

Table 6.16: Available Copies and Slot Requirements of Various Tool Types (Example 6.3).

Took Type (k)	Slot Requirement (s_k)	Number of Copies (V_k)	Compatible Machine Set ($m(k)$)
1	3	3	1,2,5
2	2	3	3,4
3	3	3	2,5
4	1	3	1,4
5	2	3	2,3

Table 6.17: Machine and Tool Magazine Capacity, and Compatibility Details (Example 6.3).

Machine (j)	Available Processing Time (C_j)	Tool Magazine Capacity (U_j)	Compatible Set of Tool Types ($\underline{t}(j)$)	Compatible Set of Operations ($\underline{B}(j)$)
1	150	8	1,4	1,2,3,9,11,14
2	200	8	1,3,5	3,4,5,8,11,12
3	200	8	2,5	1,5,7,8,9,13,14
4	150	8	2,4	2,6,7,10
5	300	8	1,3	1,4,6,10

Table 6.18: Operation Assignment Detail for the Objective of Minimizing the Total Processing Cost (Example 6.3).

Part	Operation	Machine	Tool Type	Number Assigned
1	1	1	1	2
		5	1	8
	2	4	2	5
	3	2	3	10
2	4	5	3	21
	5	2	5	14
3	6	5	1	3
	7	4	4	3
	8	3	2	6
	9	3	2	6
4	10	5	3	12
	11	2	1	4
	11	2	5	2
5	12	2	5	12
	13	3	2	4
	14	3	2	8

The total optimal processing cost = 419 units

Table 6.19: Tool Allocation and Capacity Utilization for the Objective of Minimizing the Total Processing Cost (Example 6.3).

Machine	Allocated Tool Types	Capacity Used	Slots Consumed
1	1	16	3
2	1,3,5	200	8
3	2,5	196	4
4	2,4	36	3
5	1,3	298	6

From the result depicted in the Table 6.18, it can be seen that even though no extra assignment of operations are made, the routing flexibility is obtained for operation 1 by assigning out of the total 10 number of this operation ($a_n = 5$, $b_{n,i} = 2$) 2 to machine 1 and 8 to machine 5. An important characteristic of the model of utilizing the machine capacity in best possible way can be observed from the assignment of the operation 11. For this operation, the best choice of tool is, of course, tool type 1. But due to more processing time requirement with the use of this tool as compared to that with the use of tool type 5 and little capacity left on machine 2, four number of assignments were forced to use tool type 5. From this, it can also be observed that the different occurrences of the same operation are carried out on a machine using different tool types.

It can be seen that the maximum workload is 176 units of time as compared to 298 units corresponding to machine 5 for the objective of minimizing the total processing cost (see the Tables 6.21 and 6.19).

6.4.2 Duplicate Assignments of Operations

In the models M6.1, M6.2, M6.3 and M6.4 described in the Section 6.3, a condition on unique routing and tooling has been considered. As mentioned earlier, the problem with such consideration will, of course, simplify the problem for the scheduling, but may bring down the throughput rate because of starvation of machines and blocking. This problem can be reduced if the condition on unique routing is relaxed and the leftover capacities of the machines and tool magazines are properly utilized.

Table 6.20: Operation Assignment Detail for the Objective of Minimizing the Maximum Workload on Machines (Example 6.3).

Part	Operation	Machine	Tool Type	Number Assigned
1	1	1	1	5
		3	2	5
	2	4	2	5
	3	1	4	6
		2	3	4
2	4	2	3	5
		5	3	16
	5	2	3	13
		3	5	1
3	6	5	3	3
	7	4	2	3
	8	3	2	6
	9	3	5	6
4	10	5	3	12
	11	1	1	6
5	12	2	3	2
		2	5	10
	13	3	5	4
	14	1	4	8

The maximum workload = 176 units

Table 6.21: Tool Allocation and Capacity Utilization for the Objective of Minimizing the Maximum Workload on Machines (Example 6.3).

Machine	Allocated Tool Types	Capacity Used	Slots Consumed
1	1,4	176	4
2	3,5	176	5
3	2,5	174	4
4	2	30	2
5	3	176	3

In the present section, a formulation is presented which allows multiple assignment of the operations on different machines. The assumptions made are as follows.

(i) All the occurrences of an operation of a part combined together can be assigned to more than one machine.

(ii) The occurrences of an operation of a part are performed on a machine using only one tool type. However, on different machines different tool types can be used.

For the purpose of developing the formulation, all the notations that have been introduced in the Section 6.3.1 and 6.3.2 are used. The formulation of the problem is almost the same as given in the model M6.3. The constraint (6.15) on unique routing and tooling is relaxed and is replaced by the following constraints.

$$\sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} \leq 1 \qquad n = 1, \ldots, N;\ \forall\ i \in B(n);\ \forall\ j \in M(i) \tag{6.28}$$

and

$$\sum_{j \in m(i)} \sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} \geq 1 \qquad n = 1, \ldots, N;\ \forall\ i \in B(n) \tag{6.29}$$

The constraint (6.28) ensures that all the occurrences of an operation are to be performed on a machine using only one type of a tool. The constraint (6.29), of course, allows multiple assignments of operations.

The objective of maximizing the processing flexibility can be expressed in number of ways depending upon the requirement and desirability.

Table 6.15: Resource Requirements and Related Costs for the Operations on the Parts (Example 6.3).

$N = 5$, $M = 5$ and $T = 5$

Part n	Operation $i \in B(n)$	Capable Machines $j \in m(i)$	Capable Tool Types $k \in t(i)$	Feasible Combination of Machines and Tool Types and Corresponding Processing Costs and Times $(j, k, c_{nijk}, t_{nijk})$
1	1	1,3,5	1,2	(1,1,7,8),(3,2,6,5),(5,1,3,8)
	2	1,4	2,3	(4,2,12,3)
	3	1,2	3,4	(1,4,6,8),(2,3,4,3)
2	4	2,5	3	(2,3,9,7),(5,3,2,8)
	5	2,3	3,4,5	(2,3,11,3),(2,5,4,3),(3,5,7,9)
3	6	4,5	1,3	(5,1,2,10),(5,3,9,4)
	7	3,4	2,4,5	(3,2,3,7),(3,5,4,8),(4,2,6,5),(4,4,3,7)
	8	2,3	2,4	(3,2,6,5)
	9	1,3	3,5	(3,5,4,11)
4	10	4,5	3	(5,3,5,3)
	11	2	1,5	(2,1,1,9),(2,5,2,4),(1,1,3,4)
5	12	2	3,5	(2,3,2,10),(2,5,1,7)
	13	3	1,2,5	(3,2,3,11),(3,5,9,11)
	14	1,3	2,4	(1,4,4,8),(3,2,2,7)

(c) Maximization of the sum of weighted flexibility of parts written as:

$$\text{Maximize} \left[\sum_{n=1}^{N} \omega_n \sum_{i \in B(n)} \sum_{j \in m(i)} \sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} \right],$$

where ω_n is the weight associated with part n.

(d) Maximization of the minimum weighted part processing flexibility expressed as:

$$\text{Maximize} \left[\underset{n}{\text{minimum}} \left\{ \omega_n \sum_{i \in B(n)} \sum_{j \in m(i)} \sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} \right\} \right].$$

(e) Maximization of the minimum weighted operation processing flexibility expressed as:

$$\text{Maximize} \left[\underset{n, i \in B(n)}{\text{minimum}} \left\{ \omega_n \sum_{j \in m_i} \sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} \right\} \right].$$

The objectives of maxmin type can suitably be simplified. For example, for the objective of maximizing the minimum weighted part processing flexibility, the formulation of the problem can be expressed as :

MODEL (M6.7)

Maximize R

Subject to

$$\sum_{n=1}^{N} \sum_{i \in B(n) \cap \underline{B}(j)} \sum_{k \in t(i) \cap \underline{t}(j)} t_{nijk} \cdot x_{nijk} \le C_j \qquad j = 1, \ldots, M$$

$$\sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} \le 1 \qquad n = 1, \ldots, N;$$

$$\forall\ i \in B(n);\ \forall\ j \in m(i)$$

$$\sum_{j \in M(i)} \sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} \geq 1 \qquad n = 1, \ldots, N;\ \forall\ i \in B(n)$$

$$\omega_n \sum_{i \in B(n)} \sum_{j \in m(i)} \sum_{k \in t(i) \cap \underline{t}(j)} x_{nijk} \leq R \qquad n = 1, \ldots, N$$

$$\sum_{k \in \underline{t}(j)} s_k Z_{jk} \leq U_j \qquad j = 1, \ldots, M$$

$$\sum_{k \in \underline{m}(k)} Z_{jk} \leq V_k \qquad k = 1, \ldots, T$$

$$Z_{jk} - x_{nijk} \geq 0 \qquad n = 1, \ldots, N;\ \forall\ i \in B(n);\ \forall\ j \in m(i);\ \forall\ k \in t(i) \cap \underline{t}(j)$$

$$x_{nijk} = 0 \text{ or } 1 \qquad n = 1, \ldots, N;\ \forall\ i \in B(n)\ \forall\ j \in m(i);\ \forall\ k \in t(i) \cap \underline{t}(j)$$

$$Z_{jk} = 0 \text{ or } 1 \qquad j = 1, \ldots, M;\ k \in \underline{t}(j)$$

In the above model, for assignment of an operation to more than one machine it is must that the capacity of the related machines should be enough to process all the occurrences of the operation. Because of this restriction, capacity of some of the machines may still remain unutilized which could have been utilized to provide more flexibility. Thus, an intermediate approach can be followed where number of duplicate assignments for an operation need not necessarily be an integer multiple of all the occurrences of the operation.

such resources and will depend upon the dispatching policies used.

Besides the above, two methodologies have been suggested for counting the slot requirement of the tools. The amount of slot saved for the two tools placed together depend upon the methodology used for counting slot requirement. In the first case, slot saving is always a positive number, while in the other, it is not required to consider slot saving explicitly. In the models proposed, a different method has been adopted where saving due to tool duplication is not required to be considered separately and thus avoids the complex nonlinear terms from the formulation.

CHAPTER VII

SUMMARY AND CONCLUSIONS

In the present work, the design and operational problems related with grouping of parts and machines, and allocation of machines and tools to the operations in Flexible Manufacturing System (FMS) environment have been studied, and models and solution methodologies have been presented. The FMS envisaged incorporating routing and tooling flexibility is assumed to provide alternative process plans for the parts.

Major contribution of the present thesis can be enlisted as : introduction of a new measure of commonality for grouping, formulation of generalized grouping problem using the approach for formulating p-median, p-centre and graph partitioning problem, and development of loading models incorporating different flexibility measures. The summary and conclusions for each of these are given at the end of respective chapters. However, the salient features are briefly reproduced in order to have an overall comprehensive and organized review of the present work.

The inadequacy of the popular existing similarity measures to incorporate the generality and flexibility of FMS, as considered in the present work , has been a strong motivational factor for introducing two new commonality measures, viz. Relative Requirement compatibility (RRC) and Absolute Requirement Compatibility, to be used for determining the commonality between a pair of parts, process plans, and a part (process plan) and a machine considering the bases of processing times, the number of operations, and the number of machines and tools. These measures

have been found to be quite general and also to provide the framework that can directly be used for determining the similarity based on the requirements of other resources, such as, jigs, fixtures, etc.. The relationship between RRC, Jaccard's similarity coefficient, product type similarity coefficient and cell band strength, have been established. These relations are analyzed in the light of judging the relative discriminating power of all these measures, and it has been found that RRC is the best amongst all. Numerical illustrations and examples covering a vide range of grouping problem scenarios also confirm this observation. This fact also gets stemmed from the analysis carried out on the numerical values of these commonality measures which, in addition, provides guidance and informative details for the determination of threshold values of the measures to be used as cut off points in applying certain grouping methodologies.

The generalized grouping problems with several extensions have been formulated using three distinct approaches of formulating p-median, p-centre and graph partitioning problems. The models for the grouping are proposed both for operational and design decisions. At the operational level grouping problem in FMS, the aim is to form logical groups of parts and machines, while at the design level it focuses on the physical grouping.

The existing p-median grouping formulation of Kusiak and generalized assignment formulation of Shtub are analyzed and improved in terms of lesser number of required variables and constraints. The shortcoming of these models of not resulting perfect grouping even when it is possible to achieve, has been illustrated using an example. The problem is rectified by the inclusion of an additional constraint on group disjointedness, while the use of RRC as a measure of commonality reduces the

chances of such occurrences as compared to the simple matching similarity coefficient used by these authors.

For the problems at operational level, p-median problem formulation considers the objective of maximizing the sum of similarity between member process plans and the plans that represent the corresponding groups. On the other hand, p-center formulation maximizes the minimum similarity of member process plans with plans representing process families and thus enhancing the homogeneity in group. In graph partitioning approach, instead of maximizing the similarity between a pair of a process plans, the similarity between process plans and the machines that belong to the same group is maximized. An alternative objective has also been to minimize the total inter cell movement cost. The grouping models at the design level consider the objective of minimizing the investment on machines required to result disjoint groups, and are proposed using the approaches of p-median and group partitioning problems formulation. These approaches also consider certain other important factors related with the design and operations of the system, such as, the limits on the size of the part families and machine cells, the number of machines of each type and the capacity of machines, thus making the decisions more practical and amenable for application in manufacturing environment such as FMS. On the other hand, consideration of these features of grouping makes the problem harder to solve. With this in view, heuristic approaches for some of these models have been proposed.

The heuristics developed for p-median and p-centre grouping formulations determine the groups in an hierarchical manner. First, p process plans are determined in a greedy manner to represent the required number of process families. Using this

are carried out. Next, the efforts are made to improve the objective function value by perturbing selectively the plans representing process families. The heuristic based on graph partitioning approach forms the groups simultaneously. In addition, the heuristics have also been proposed which are based on the concepts of Lagrangian relaxation and use subgradient optimization procedure for seeking the optimal solution.

In the process of development of the models and solution methodologies, it is found that similarity coefficient based approaches are more suitable for forming the groups hierarchically in case of simple grouping where each part has a single process plan. In generalized grouping, such as in FMS, they do not seem to serve the purpose much. Graph partitioning formulation when used for hierarchical group formation may lead to an inferior grouping solution as it may not account for the disjointedness of the groups in hierarchical approach of group determination.

Some of the models proposed for grouping intrinsically integrate the loading problems for providing solution to assignments of operations to machines. Additional allocation decisions, such as of tools, can also be determined using the loading models presented in this thesis.

The proposed loading models are sufficiently general incorporating the flexibility in performing an operation using various combinations of tool and machine types. The models are free of nonlinear terms required to represent the savings in the total slot requirement due to tool duplication and overlapping. The savings because of tool duplication are accommodated in the proposed formulation itself and no separate expressions are required. The savings due to tool overlap are shown to depend

tools. Out of the two proposed methods, one suitably takes care of the savings due to tool overlap. In fact, the computation of slot requirements is done in a way that does not require explicit consideration of savings due to overlap. The other proposed method shows a systematic way for counting the slot requirements of tools and also for determining the total savings due to tool overlap, but does require explicit mention of the related expression.

Generally, incorporation of flexibility into consideration makes the loading problem complex to solve and analyze, but helps to increase the throughput rate and to utilize the resources in a better manner. With this view, certain quantitative measures for the flexibility of processing a part have been proposed, and loading models are presented incorporating such measures. Numerical examples presented exhibit the advantages achieved.

In addition, the issues related with loading problems in FMS environment have also been discussed. The issues are the aspects of flexibility and the problem parameters that are to be considered for a better decision making. It has been shown that out of tools, pallets, jigs and fixtures, important resource to be considered for solving the operations assignment problems is the tool along with machines. The allocation of other resources is proposed to be considered at the stage of real-time scheduling.

In the present study, comparison between the formulations of the grouping problems obtained using different approaches have been made only in terms of the number of involved decision variables and constraints. However, an analysis needs to be made for judging the quality of the solutions obtained using different approaches for the same grouping situation. Besides, an analysis

the models in terms of the CPU requirement and the quality of the solution.

The methodologies proposed in the thesis for determining logical grouping, and loading of tools and operations on machines, can be of great advantage for finding these decisions even in the case of integration of GT with materials requirement planning (MRP). However, a batching algorithm that acknowledges the flexibility of FMS and also for performing the operations of the parts, the capacity of the machines, requirement of other resources such as cutting tools etc., needs to be developed for finding from a master production schedule a cost and production efficient operational schedule using the concepts of GT. The problem may be complex to model and analyze mathematically. For such a situation, the use of Decision support system, Expert system and simulation techniques could be advantageous.

Further, the grouping models can also be used for carrying out parametric analysis to view the economic and organizational effects of the total number of groups, limits on the size of groups and the total number of machines of various types. The analysis will result numerous possible answers to the grouping problem, and the one that provides maximum satisfaction and meets the objective most, can be chosen for implementation.

REFERENCES

Agrawal, A.K. and Shanker, K. (1988), "Computation of Tool Slot Requirement," A working paper, Indian Institute of Technology, Kanpur, India.

Al-Qattan, I. (1990), "Designing Flexible Manufacturing Cells Using a Branch and Bound Method," International Journal of Production Research, Vol. 28, No. 2, 325-336.

Anderberg, M.R. "Cluster Analysis for Applications," Academic Press, New York, 1973.

Askin, R.G. and Chiu, K.S. (1987), "A Graph Partitioning Procedure for Machine Assignment and Cell formation in Group Technology," Working Paper No. 87-009, College of Engineering, Univ. of Arizona, Tucson, Arizona.

Askin, R.G. and Subramanian, S.P. (1987), "A Cost Based Heuristic for Group Technology Configuration," International Journal of Production Research, Vol. 25, No. 1, 101-113.

Ballakur, A. and Steudel, H.J. (1987), "A Within Cell Utilization Based Heuristic for Designing Cellular Manufacturing Systems," International Journal of Production Research, Vol. 25, No. 5, 639-665.

Berrada, M. and Stecke, K. (1986), "A Branch and Bound Approach for Machine Load Balancing in Flexible Manufacturing Systems," Vol. 30, No. 10, 1316-1335.

Burbidge, J.L. (1963), "Production Flow Analysis," Production Engineer," Vol. 42, No. 12, 742-752.

Burbidge, J.L. (1971), "Production Flow Analysis," Production Engineer, Vol. 50, No. 4/5, 139-152.

Burbidge, J.L. (1977), "A Manual Method of Production Flow Analysis," Production Engineer, Vol. 56, 34-38.

Burbidge, J.L. (1988), "Operations Scheduling with GT and PBC," International Journal of Production Research," Vol. 26, No. 3, 429-442.

Chandrasekharan, M.P. and Rajagopalan, R. (1986), "An Ideal Seed Nonhierarchical Clustering Algorithm for Cellular Manufacturing," International Journal of Production Research, Vol. 24, No. 2, 451-464.

Chandrasekharan, M.P. and Rajagopalan, R. (1986), "MODROC : An Extension of Rank Order Clustering for Group Technology," International Journal of Production Research, Vol. 24, No. 5,

Vol. 12, No. 2, 131-142.

King, J.R. (1980), "Machine-Component Grouping in Production Flow Analysis: An Approach Using Rank Order Clustering," International Journal of Production Research,Vol. 18, No. 2, 213-232.

King, J.R. and Nakornchai, V. (1982), "Machine-Component Group Formation in Group Technology: Review and Extension," International journal of Production Research, Vol. 20, No. 2, 117-123.

Kiran, A.S. and Tansel, B.C. (1985), "A Framework for Flexible Manufacturing System," Proceedings of Annual International Industrial Engineering Conference, 446-450.

Krishna, S.M. and Datar, A.D. (1988), "An Improved Algorithm for Machine-Component Cell Formation in Group Technology," 13th AIMTDR Conference, Jadavpur University, India, I-01 - I-09.

Kumar, K.R., Kusiak, A. and Vannelli, A. (1986), "Grouping of Parts and Components in Flexible Manufacturing Systems," European Journal of Operational Research, Vol. 24, 387-397.

Kumar, K.R. and Vannelli, A. (1987), "Strategic Subcontracting for Efficient Disaggregated Manufacturing," International Journal of Production Research, Vol. 25, No. 12, 1715-1728.

Kumar, P., Tewari, N.K. and Singh, M. (1987), "Joint Consideration of Grouping and Loading Problems in a Flexible Manufacturing Systems," Working Paper No. 87087, IIT Delhi, India.

Kusiak, A. (1984), "The Part Families Problem in Flexible Manufacturing Systems," Working Paper No. 06/84, Halifax, Nova Scotia.

Kusiak, A. (1985), "Loading Models in Flexible Manufacturing Systems," Raouf, A. and Ahmad, S.I. (Ed.), Manufacturing Research and Technology - I, Elsevier.

Kusiak, A. (1985), " Flexible Manufacturing Systems : A Structural Approach," International Journal of Production Research, Vol. 23, No. 6, 1057-1073.

Kusiak, A. (1985), "Planning of Flexible Manufacturing Systems," Robotica, Vol. 3, 229-232.

Kusiak, A., Vannelli, A. and Kumar, K.R. (1985), "Grouping Problem in Scheduling Flexible Manufacturing Systems," Robotica, Vol. 3, 245-252.

Kusiak, A. (1987), "The Generalized Group Technology Concept," International Journal of Production Research, Vol. 25, No. 4, 561-569.

Kusiak, A. (1987), "Artificial Intelligence and Operations Research in Flexible Manufacturing Systems," INFOR, Vol. 25, No.

Kusiak, A. (1987), "Artificial Intelligence and Operations Research in Flexible Manufacturing Systems," INFOR, Vol. 25, No. 1, 2-12.

Kusiak, A. (1987), "An Expert System for Group Technology," Industrial Engineering, No. 10, 56-61.

Kusiak, A. (1988), "EXGT-S : A Knowledge Based System for Group Technology," International Journal of Production Research, Vol. 26, No. 5, 887-904.

Lashkari, R.S., Dutta, S.P. and Padhye, A.M. (1987), "A New Formulation of Operation Allocation Problem in Flexible Manufacturing Systems : Mathematical Modeling and Computational Experience," International Journal of Production Research, Vol. 25, No. 9, 1267-1283.

Lawande, N.P.P. and Datar, A.D. (1988), "An Algorithm for Machine-Component Analysis for Cell Formation in Group Technology," Industrial Engineering Journal, Vol. 17, No. 12, 14-22.

Mazzola, J.B., Neebe, A.W. and Dunn, C.V.R. (1987), "Production Planning of a Flexible Manufacturing System in a Material Requirement Planning Environment," Working Paper No. 8608, Fuqua School of Business, Duke University, Aug. 1986, Revised 1987.

McAuley, J. (1972), "Machine Grouping for Efficient Production," Production Engineer, Vol. 51, No. 2, 53-57.

Minieka, E. (1970), "The m-centre Problem," SIAM Rev., Vol. 12, 138-139.

Minieka, E. (1977), "The Centres and Medians of a Graph," Vol. 25, No.4, 641-650.

Mosier, C. and Taube, L. (1985), "Weighted Similarity Measure Heuristics for the Group Technology Machine Clustering Problem," OMEGA, Vol. 13, No. 6, 577-583.

Mosier, C.T. (1989), "An Experiment Investigating the Application of Clustering Procedures and Similarity Coefficients to the GT Machine Cell Formation Problem," International Journal of Production Research, Vol. 27, No. 10, 1811-1835.

Na, Y.K., Han, M.H. and Hogg, G.L. (1987), "Tool Loading and Job Assignment in an FMS," A Working Paper, Dept. of Industrial Engineering, Texas A & M University, College Station, USA.

O'grady, P.J. and Menon, U. (1987), "Loading a Flexible Manufacturing System," International Journal of Production Research, Vol. 25, No. 7, 1053-1068.

Parker, R.G. and Rardin, R.R. "Discrete Optimization," Academic Press, Boston, 1989.

Rajagopalan, R. and Batra, J.L. (1975), "Design of Cellular Production Systems: A Graph Theoretic Approach," International

Remmerwaal, J.L., Furth, B. and Roubos, P. (1983), "Mischedule - A Group Technology Approach to Sequencing of Jobs," Annals of the CIRP, Vol. 32, No. 1, 389-392.

Ross, G.T. and Soland, R.M. (1975), "A Branch and Bound Algorithm for the Generalized Assignment Problem," Mathematical Programming, Vol. 8, 91-103.

Roy, N. and Sengupta, D.K. (1989), "Towards a Practical Rank Order Clustering Algorithm for Machine-Component Group Formation - A Software Development," Proceedings of International Conference on Engineering Software, IIT Delhi, India, 161-165.

Saraf, R.K. (1989),"Computerized Production Planning System for an Electronic Manufacturing Company," Unpublished M. Tech. Thesis, I.I.T. Kanpur, India.

Sato, N., Ignizio, J.P. and Ham, I. (1978)," GT and MRP : An Integrated Methodology for Production Control," Annals of the CIRP, Vol. 27, No. 1, 471-473.

Seifoddini, H. (1988), "Incorporation of the Production Volume in Machine Cells Formation in Group Technology Applications," Recent Development in Production Research, A. Mital (Ed.), Elsevier Science Publishers B.V., Amsterdam.

Seifoddini, H. (1989a), "Single Linkage Versus Average Linkage Clustering in Machine Cells Formation Applications," Computers and Industrial Engineering, Vol. 16, No. 3, 419-426.

Seifoddini, H. (1989b), "Comparison Between the Similarity Coefficient Method and Machine-Component Group Analysis in Group Technology Applications," Computers and Industrial Engineering, Vol. 17, No. 1-4, 609-613.

Seifoddini, H. (1989c), "A Note on the Similarity Coefficient Method and the Problem of Improper Machine Assignment in Group Technology," International Journal of Production Research, Vol. 27, No. 7, 1161-1165.

Seifoddini, H. and Wolfe, P.M. (1986), "Selection of a Threshold Value Based on the Material Handling Cost in Machine-Component Grouping," I.I.E. Transactions, Vol. 19, No. 3, 266-270.

Sarin, S.C. and Chen, C.S. (1987), "The Machine Loading and Tool Allocation Problem in a Flexible Manufacturing System," International Journal of Production Research, Vol. 25, No. 7, 1081-1094.

Shah, J.J and Bhatnagar, A.S. (1989), "Group Technology Classification from Feature-Based Geometric Model," Manufacturing Review, Vol. 2, 204-213.

Shanker, K. and Srinivasulu (1989), "Some Solution Methodologies for Loading Problems in a Flexible Manufacturing Systems,"

International Journal of Production Research, Vol. 27, No. 6, 1019-1034.

Shanker, K. and Tzen, Y.J. (1985), "Loading and Dispatching Problem in Random Flexible Manufacturing System," International Journal of Production Research, Vol. 23, 579-595.

Shanker, K. and Zhu, W. (1987), "Notes on King's Rank Order Clustering Algorithm in Group Technology," A Working Paper, AIT Bangkok, Thailand.

Shapiro, J.F. (1979), "A Survey of Lagrangean Techniques for Discrete Optimization," Annals of Discrete Mathematics, Vol. 5, 113-138.

Shtub, A. (1989), "Modeling Group Technology Cell Formation as a Generalized Assignment Problem," International Journal of Production Research, Vol. 27, No. 5, 775-782.

Srinivasan, G., Narendran, T.T. and Mahadevan, B. (1990), "An Assignment Model for the Part-Families Problem in Group Technology," International Journal of Production Research, Vol. 28, No. 1, 145-152.

Stecke, K.E. (1983), "Formulation and Solution of Nonlinear Integer Production Planning Problems for Flexible Manufacturing Systems," Management Science, Vol. 29, No. 3, 273-288, 1983.

Stecke, K.E. (1986), "A Hierarchical Approach to Solving Machine Grouping and Loading Problems of Flexible Manufacturing Systems," European Journal of Operations Research, Vol. 24, 369-378.

Stecke, K.E. and Browne, J. (1985), "Variations in Flexible Manufacturing System According to Relevant Types of Automated Material Handling," Material Flow, Vol. 2, 179-185.

Stecke, K.E. and Solberg, J.J. (1981), "Loading and Control Policies for a Flexible Manufacturing System," International Journal of Production Research, Vol. 19, No. 5, 481-490.

Stecke, K.E. and Talbot, F.B. (1983), "Heuristic Loading Algorithms for Flexible Manufacturing Systems," Proceeding of Seventh International Conference on Production Research, Windsor, Ontario, Canada, 22-24.

Studel, H.J. and Ballakur, A. (1987), "A Dynamic Programming Based Heuristic for Machine Grouping in Manufacturing Cell Formation," Computers and Industrial Engineering, Vol. 12, No. 3, 215-222.

Taboun, S.M., Dutta, S.P. and Richardson, K. (1989), "Comparison and Evaluation of Cellular Manufacturing Design Techniques," Computers and Industrial Engineering, Vol. 17, Nos. 1-4, 603-608.

Tremblay, J.P. and Sorensan, P.G.. "An Introduction to Data Structures with Applications," McGraw-Hill International Edition, 1984.

Tuffentsammer, K. and Arndt, G. (1983), "The Influence of NC Flexibility on the Concept of Group Technology," Annals of the CIRP, Vol. 32, No. 1, 411-415.

Vannelli, A. and Kumar, K.R. (1986), "A Method for Finding Minimal Bottleneck Cells for Grouping Part-Machine Families," International Journal of Production Research, Vol. 24, No. 2, 387-400.

Ventura, J.A., Chen, F.F. and Leonard, M.S. (1987), "Grouping Parts and Tools in FMS Production Planning," Working Paper No. 8710904, Dept. of Industrial Engineering, University of Missouri - Columbia, Columbia.

Waghodekar, P.H. and Sahu, S. (1983), "Group Technology - A Research Bibliography," OPSEARCH, Vol. 20, No. 4, 225-249.

Waghodekar, P.H. and Sahu, S. (1984), "Machine-Component Cell Formation in Group Technology MACE," International Journal of Production Research, Vol. 22, No. 6, 937-948.

Wei, J.C. and Kern, G.M. (1989), "Commonality Analysis : A Linear Cell Clustering Algorithm for Group Technology," International Journal of Production Research, Vol. 27, No. 12, 2053-2062.

Wilson, J.M. (1989), "An Alternative Formulation of the Operation Allocation Problem in Flexible Manufacturing Cell," International Journal of Production Research, Vol. 27, No. 8, 1405-1412.

Xu, H. and Wang, H.P. (1989), "Part Family Formation for GT Applications Based on Fuzzy Mathematics," International Journal of Production Research, Vol. 27, No. 9, 1637-1651.

Yao, D. (1985), "Material and Information Flows in Flexible Manufacturing Systems," Material Flow, Vol. 2, 143-149.

BIBLIOGRAPHY

Ang, C.L. and Willey, P.C.T. (1984), "A Comparative Study of the Performance of Pure and Hybrid GT Manufacturing Systems Using Computer Simulation Techniques," International Journal of Production Research, Vol. 22, No. 2, 193-233.

Banerjee, A. and Flynn, B.B. (1987), "A Simulation Study of Some Maintenance Policies in a GT Shop," International Journal of Production Research, Vol. 25, No. 11, 1595-1609.

Burbidge, J.L. (1979), "Group Technology in the Engineering Industry," Mechanical Engineering Publications, London.

Carrie, A.S. (1973), "Numerical Taxonomy Applied to Group Technology and Plant Layout," International Journal of Production Research, Vol. 11, No. 4, 399-416.

Chakrabarty, A.K. and Shtub, A. (1987), "Capacity Cost and Scheduling Analysis for a Multi Product Manufacturing Cell," International Journal of Production Research, Vol. 25, No. 8, 1143-1156.

Chu, C.H. (1989), "Cluster Analysis in Manufacturing Cellular Formation," OMEGA, Vol. 17, No. 3, 289-295.

Connoly, R. and Sabberwal, A.J.P. (1973), "Group Manufacture," Annals of the CIRP, Vol. 22, No. 1, 147-148.

Dale, B. and Willey, P. (1980), "How to Predict the Benefits of Group Technology," Production Engineer, Vol. 59, No. 2, 51-54.

Dewitte, J. (1980), "The Use of Similarity Coefficients in Production Flow Analysis," International Journal of Production Research, Vol. 18, No. 4, 503-514.

Durie, F.R.E. (1970), "A Survey of Group Technology and its Potential for Uses in UK," Production Engineer, Vol. 49, No. 2, 51-61.

Edwards, G.A.B. (1971), "The Family Grouping Philosophy," International Journal of Production Research, Vol. 9, No. 3, 337-352.

Edwards, G.A.B. and Koenigsberger, F. (1973), "Group Technology, the Cell System and Machine Tools," Production Engineer, Vol. 52, No. 7/8, 249-256.

Fazakerley, G.M. (1974), "Group Technology - Social Benefits and Social Problems," Production Engineer, Vol. 53, No. 10, 383-386.

Flynn, B.B. and Jacobs, F.R. (1986), "A Simulation Comparison of Group Technology with Traditional Job Shop Manufacturing," International Journal of Production Research, Vol. 24, No. 5, 1171-1192.

Gallagher, C.C., Grayson, T.J. and Phillips, A. (1971), "A Review of the Development and Application of Group Technology at the University of Birmingham," International Journal of Production Research, Vol. 9, No. 2, 229-237.

Grayson, T.J. (1978), "Some Research Findings on Evaluating the Effectiveness of Group Technology," International Journal of Production Research, Vol. 16, No. 2, 89-102.

Greene, T.J. and Sadowski, R. P. (1983), "Cellular Production Control," Journal of Manufacturing Systems," Vol. 12, No. 2, 137-144.

Gupta, R.M. and Tompkins, J.A. (1982), "An Examination of the Dynamic Behaviour of Part-Families in Group Technology," International Journal of Production Research, Vol. 20, No. 1, 73-86.

Hyer, N.L. and Wemmerlöv, U. (1984), "Group technology and Productivity," Harvard Business Review, Vol. 621, No. 4, 140-149.

Hyer, N.L. and Wemmerlöv, U. (1985), "Group Technology Oriented Coding Systems Structures, Applications and Implementation," Production and Inventory Management, Vol. 26, No. 2, 55-78.

Knight, W.A. (1974), "Part Family Methods for Bulk Metal Forming," International Journal of Production Research, Vol. 12, No. 2, 209-231.

Knight, D.O. and Wall, M.L. (1989), "Using Group Technology for Improving Communication and Coordination Among Teams of Workers in Manufacturing Cells," Industrial Engineering, Vol. 21, No. 1, 28-34.

Koenigsberger, F. (1972), "The Use of Group Technology in the Industries of Various Countries," Annals of the CIRP, Vol. 21, No. 2, 209-214.

Kumar, C.S. and Chandrashekharan, M.P. (1990), "Grouping Efficacy: A Quantitative Criterion for Goodness of Block Diagonal Forms of Binary Matrices in Group Technology," International Journal of Production Research, Vol. 28, No. 2, 233-243.

Kusiak, A. (1983), "Part Families Selection Model for Flexible Manufacturing Systems," Proceedings of 1983 Annual Industrial Engineering Conference, Institute of Industrial Engineers, 575-579.

McCormick, W.T. (Jr.), Scweitzer, P.J. and White, T.W. (1972), "Problem Decomposition and Data Reorganization by a Clustering Technique," Operations Research, Vol. 20, 993-10009.

Mills, M.C. (1986), "Using Group Technology, Simulation and Analytic Modeling in the Design of a Cellular Manufacturing Facility," Proceedings of the 1986 Winter Simulation Conference, Wilson, J., Henriksen, J. and Roberts, S. (Eds.), 657-660.

Mitrafanov, S.P. "Scientific Principles of Group Technology," Leningrad, 1959.

Mosier, C. and Taube, L. (1985), "The Facets of Group Technology and Their Impacts on Implementation - A State of the Art Survey," OMEGA, Vol. 13, No. 5, 381-391.

Nisanci, H.I. and Sury, R.J. (1981), "An Application of Group Technology Concepts in Shoe Manufacturing," International Journal of Production Research, Vol. 19, No. 3, 267-275.

Opitz and Weindahl, H.P. (1971), "Group Technology and Manufacturing Small and Medium Quantity Production," International Journal of Production Research, Vol. 9, No. 1, 181-203.

Peklenik, J. and Grum, J. (1980), "Investigation of Computer Aided Classification of Parts," Annals of the CIRP, Vol. 29, No. 1, 319-323.

Pullen, R.D. (1976), "A Survey of Cellular Manufacturing Cells," Production Engineer, Vol. 55, No. 9, 451-454.

Purcheck, G. (1985), "Machine Component Group Formation - A Heuristic Method for Flexible Production Cells and Flexible Manufacturing Systems," International Journal of Production Research, Vol. 21, No. 5, 911-943.

Rosenblatt, M.J. and Finger, N. (1983), "An Application of a Grouping Procedure to a Multi-Item Production System," International Journal of Production Research, Vol. 21, No. 2, 223-229.

Sasani, F. (1990), "A Simulation Study on Performance Improvement of Group Technology Cells," International Journal of Production Research, Vol. 28, No. 2, 293-300.

Seifoddini, H. and Wolfe, P.M. (1986), "Selection of a Threshold Value Based on the Material Handling Cost in Machine-Component Grouping," I.I.E. Transactions, Vol. 19, No. 3, 266-270.

Torrance, J. (1972), "Correspondence to the Editor," Production Engineer, Vol. 51, No. 11, 401-402.

Warnecke, H.J., Steinhilper, R. and Roth, H.P. (1986), "Developments and Planning for FMS - Requirements, Examples and Experiences," International Journal of Production Research, Vol. 24, No. 4, 763-772.

Wemmerlöv, U. and Hyer, N.L. (1989), "Cellular Manufacturing in the US Industry: a Survey of Users," International Journal of Production Research, Vol. 27, No. 9, 1511-1530.

m : a machine type

Parameters

p = the total number of groups to be performed

q = the total number of process plans over all the parts

M = the total number of machine types

N = the total number of parts

B(n) = set of operations to be performed on part n

L = a large number (= $\sum_{n=1}^{N} |B(n)|$)

$\underline{B}(m)$ = set of operations that can be performed on machine type m

N_m = set of machines of type m

M_L = the minimum number of machines that must be assigned to a machine cell

M_U = the maximum number of machines that can be assigned to a group

M_T = the total number of machines that can be accommodated by the production system

P_L = the minimum number of parts that must be assigned to a group

P_U = the maximum number of parts that can be assigned to a group

T_m = capacity of machine type m

$a_{n,i}$ = intercell movement cost for operation i of part n

C(i) = set of machines on which operation i can be carried out

a_{nim} = a number representing strength of association between machine type m and operation i of part n

As mentioned earlier, it may not be possible to assign all the operations of all the parts to the machines belonging to the same group because of incapability of the machines to process them or because of the excess loads on them. In situations like this, intercell movements are unavoidable and the constraint (5.3) has to be modified to:

$$\sum_{m \in C(i)} x_{nimk} \le y_{nk} \qquad n = 1, \ldots, N; \quad \forall\, i \in B(n); \quad k = 1, \ldots, p. \tag{5.4}$$

(ii) <u>Consistency Between Operation and Machine Assignments</u>

The constraint that if an operation is assigned to a machine, then the machine should be available in the group to which the part corresponding to that operation belongs, can be expressed as:

$$\sum_{n=1}^{N} \sum_{i \in B(n) \cap \underline{B}(m)} x_{nimk} \le L\, Z_{mk} \qquad \forall\, m = 1, \ldots, M; \quad k = 1, \ldots, p. \tag{5.5}$$

(iii) <u>Part Assignment</u>

The constraint that each part is assigned to one and only one group can be written as:

$$\sum_{k=1}^{p} y_{nk} = 1 \qquad n = 1, \ldots, N. \tag{5.6}$$

(iv) <u>Total Number of Machines of Each Type</u>

The constraint that total number of machines of a type assigned to various cells should be less than or equal to its available number is:

$$\sum_{k=1}^{p} Z_{mk} \le |N_m| \qquad m = 1, \ldots, M. \tag{5.7}$$

$$+ \sum_{n=1}^{N} \sum_{k=1}^{p} w_{mk} \left(L\, Z_{mk} - \sum_{n=1}^{N} \sum_{i \in B(n) \cap \underline{B}(m)} x_{nimk}\right)$$

Subject to: (5.4), (5.6), (5.7), (5.8), (5.9), (5.10), (5.11), (5.13), (5.14), (5.15) and

$$w_{mk} \geq 0 \qquad m = 1, \ldots, M;\ k = 1, \ldots, p \qquad (5.16)$$

On rearranging the terms in the objective function of the above formulation, following is obtained.

(D1)

$$Z_D(w) = \text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} \sum_{k=1}^{p} (a_{nim} - w_{mk}) x_{nimk}$$

$$+ L \sum_{m=1}^{M} \sum_{k=1}^{p} w_{mk}\, Z_{mk}$$

Subject to: (5.4), (5.6), (5.7), (5.8), (5.9), (5.10), (5.11), (.13), (5.14), (5.15) and (5.16).

The above problem can be broken into following two subproblems P_a and P_b for known values of w_{mk}'s.

(P_a)

$$Z_1 = \text{Maximize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{m \in C(i)} \sum_{k=1}^{p} (a_{nim} - w_{mk}) x_{nimk}$$

Subject to: (5.4), (5.6), (5.10), (5.11), (5.13) and (5.14).

(P_b)

$$Z_2 = L\ \text{Maximize} \sum_{m=1}^{M} \sum_{k=1}^{p} w_{mk}\, Z_{mk}$$

Subject to: (5.7), (5.8), (5.9) and (5.15).

It can be observed that for the known values of dual variables w_{mk}'s the subproblem P_a is effectively the same as the

$$\sum_{n \in G_P'(K)} \sum_{i \in B'(n) \cap \underline{B}'(m)} t_{nim} \, \underline{x}_{nimk} \le T_m' \, Z_{mk}$$

$$\forall \; m \in G_M'(k)$$

5.4.3 An Alternative Approach for Grouping

In this section, a different approach is taken for solving the grouping problem. However, the basic idea behind the grouping remains the same as used in the previous section. The approach aims at maximizing the sum of association between operations of the parts and machines that are assigned to the same group. The heuristic presented in the Section 5.4.3.1 is to be used for the generalized grouping situation. The heuristic with improvements and additional features is presented in the Section 5.4.3.2, but for the simple grouping situation.

5.4.3.1 GENERALIZED GROUPING PROBLEM

The problem considered for grouping in this section does not include the constraints on size of the group and on load of machines. Moreover, the groups are formed in a natural way without considering the restriction on its total number. For this situation, the heuristic proposed, to be called as GAO (Grouping when Alternatives exist for carrying out Operation), uses two phase procedure. In the first phase, groups are formed in a natural way. The second phase is to be used for further refinement and final adjustment of groups.

HEURISTIC

The two phases of the heuristic are as follows.

Phase I

Step 0: Let P be the set of parts and M be the set of machine types. Find for each alternative process plan of each

the total number of unassigned copies of the machine type.

From the machine cell formed, remove all those machines which satisfy both of the following conditions:

(i) Machines on which the requirements of the parts in the present part family are not more than their corresponding admissibility factors, and

(ii) Machines for which the rejectability factors are greater than or equal to their corresponding admissibility factors.

Similarly, remove all those parts from the presently formed part family whose more than half requirement cannot be met by the present machine cell. Continue this process until no removal takes place or when either of the machine cell or part family becomes empty.

Step 3: If both the machine cell and the part family are not empty, set K = 1 and go to Step 5. Otherwise, follow the next step.

Step 4: Take out the part from the sequence. If the sequence has some parts for consideration, then make the part family and the machine cell empty and with the remaining parts in the sequence go to Step 1.

Otherwise, with the complete sequence determined in the Step 0 follow Step 1 onwards but replacing the conditions of more than by greater than or equal to in the Step 1, 2 and 6, and that of greater than or equal to by greater than in the Step 2 used for selection of

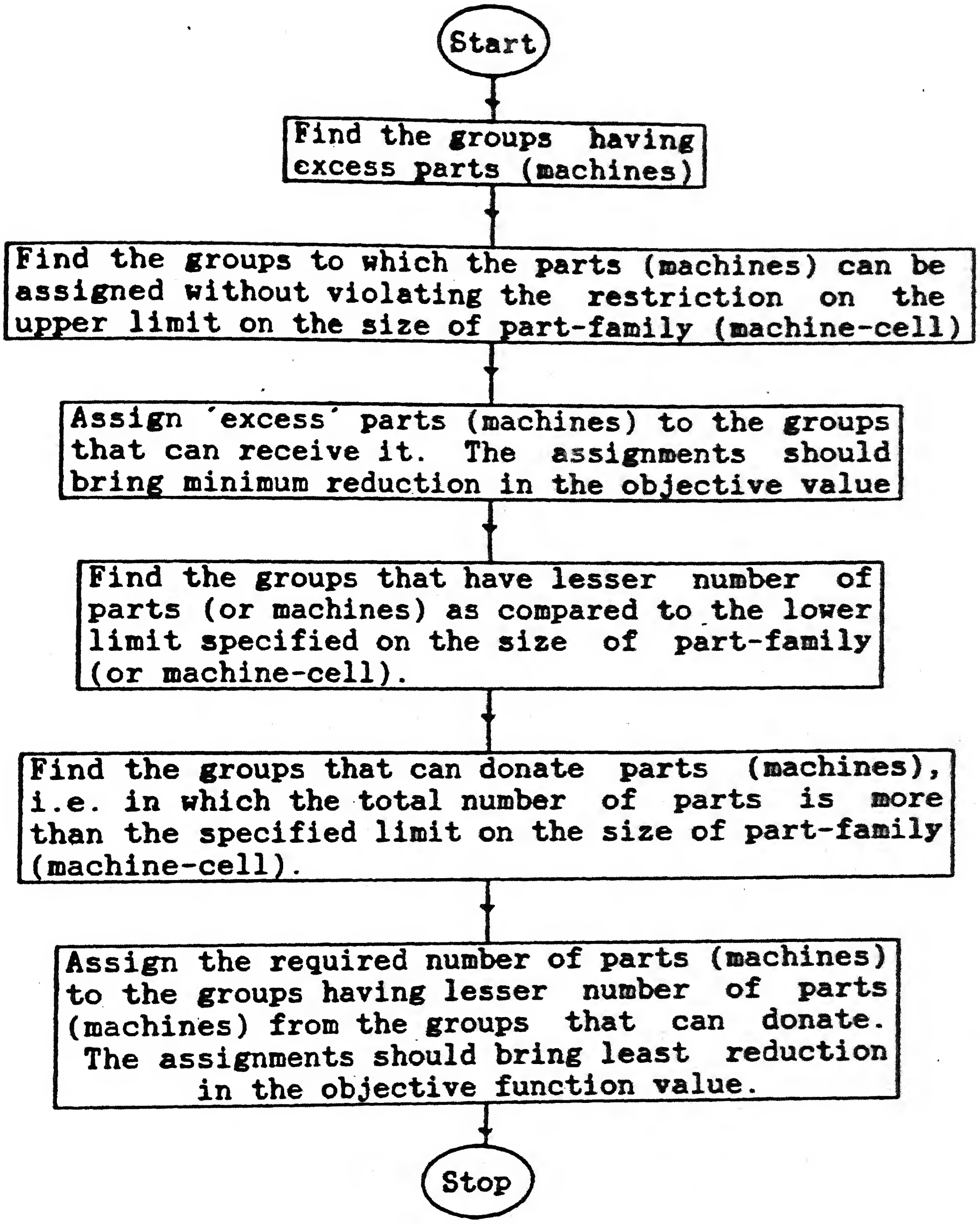


Figure 5.13: Flowchart Showing Details for Rearrangement of Parts and Machines among the Groups for Satisfying the Limits on Group Size.

Table 5.13: Operation Assignment Details for the Case When Machine Capacity Constraint is Included and the Restriction on Group Disjointedness is Relaxed.

Objective function value = 91

Part	Operation	Assigned Machine
1	1	1
	2	–
	3	3
2	4	1
	5	6
3	6	2
	7	2
4	8	4
	9	5
	10	2
5	11	4
	12	5
6	13	4
	14	5

Table 5.14: Group Configuration Details for the Case When Machine Capacity Constraint is Included and the Restriction on Group Disjointedness is Relaxed.

Group #	Part-Family	Machine-Cell
1	{1, 3, 6}	{1, 2}
2	{2, 4, 5}	{3, 4, 5, 6}

The same problem when solved to determine the disjoint groups requiring minimum investment on machines with $|N_m| = 2 \ \forall \ m$, results solutions as are shown in Tables 5.15 and 5.16. The problem is an example for illustrating the model M5.6.

c_{nijk} = cost of processing all the occurrences of operation i on all the units of part n using machine j and tool type k ($= a_n . b_{ni} . t_{nijk}$)

t_{nijk} = time required for processing a single occurrence of operation i of part n using machine j & tool type k

T_{nijk} = time required for processing all the occurrences of operation i on all the units of part n using machine j and tool type k ($= a_n . b_{ni} . t_{nijk}$)

Decision Variables

L = maximum workload on machines

O_j = overload on machine j

$$Z_{jk} = \begin{cases} 1 & \text{if tool type k is assigned to machine j} \\ 0 & \text{otherwise} \end{cases}$$

$$x_{nijk} = \begin{cases} 1 & \text{if all the occurences of operation i of part n are performed on machine j using a tool of type k} \\ 0 & \text{otherwise} \end{cases}$$

X_{nijk} = the number of occurrences of operation i of part n that are to be performed on machine j using a tool of type k

The operations of the various parts are indexed sequentially. For example, for part 1, the operations are indexed from 1 to $|B(1)|$, for part 2 from $|B(1)|+1$ to $|B(1)| + |B(2)|$, and so.

6.3 LOADING MODELS FOR UNIQUE ROUTING AND TOOLING

The loading problem for the unique routing and tooling assumes that all the units of a part follow the same route and an operation is processed using only one combination of a tool and a machine. This assumption characterizes the general problem situation. Additional details of the problem depend upon the nature of

then the constraint will be:

$$\sum_{k=1}^{T} s_{jk} Z_{jk} \le U_j \qquad j = 1, \ldots, M$$

where s_{jk} will represent the slot requirement of the tool type k on machine j.

(iv) Copies of each tool type

Allocated number of each tool type should not exceed the available number. The constraint can be represented in the following form.

$$\sum_{j=1}^{M} Z_{jk} \le V_k \qquad k = 1, \ldots, T \qquad (6.5)$$

(v) Tool allocation for the assigned operation

The constraint that to the machines all the tools are allocated that are needed to perform the operations assigned to these machines, is represented as:

$$x_{nijk} = 1 \implies Z_{jk} = 1 \qquad i = 1, \ldots, N;\ \forall\ i \in B(n);$$
$$j = 1, \ldots, M;\ \forall\ k \in t(i).$$

The above constraint is simplified using the scheme suggested by Garfinkel and Nemhauser (1980) and is written in the usual form of the constraints as shown below.

$$x_{nijk} \le \gamma_{nijk} \qquad i = 1, \ldots, N;\ \forall\ i \in B(n);\ j = 1, \ldots, M;\ \forall\ k \in t(i) \qquad (6.6)$$

$$z_{jk} - 1 \ge (1 - \gamma_{nijk})(-1) \qquad i = 1, \ldots, N;\ \forall\ i \in B(n);\ j = 1, \ldots, M;\ \forall\ k \in t(i) \qquad (6.7)$$

where $\gamma_{nijk} = 0$ or 1 $\qquad i = 1, \ldots, N;\ \forall\ i \in B(n);$

$j = 1, \ldots, M;\ \forall\ k \in t(i).$

$$z_{jk} = 0 \text{ or } 1 \qquad j = 1, \ldots, M \text{ and } k \in \underline{t}(j) \tag{6.20}$$

MODEL (M6.3)

The formulation of the problem is summarized below:

$$\text{Minimize} \sum_{n=1}^{N} \sum_{i \in B(n)} \sum_{j \in m(i)} \sum_{k \in t(i) \cap \underline{t}(j)} c_{nijk} \, x_{nijk}$$

Subject to: (6.14), (6.15), (6.16), (6.17), (6.18), (6.19) and (6.20).

EXAMPLE

An example with the details of the problem parameters given in Tables 6.8, 6.9 and 6.10 is considered. It covers the range from high flexibility to no flexibility in routing and tooling for operations. For example, for performing operation 1 machines 1, 3 and 5 can be used, and the tools 1 and 2; whereas for operation 10 only tool type 3 can be used, and for performing operation 11 only machine 2 can be used. It can also be noticed that all the combinations of those tools and machines that individually can be used for performing an operation, may not be feasible because of the compatibility problem of the tool and machine. For example, out of the total six combinations of machines and tools, the feasible combinations of machines and tools for operation 1 of part 1 are (1,1), (3,2) and (5,1) only.

The resulting assignment of operations are given in Table 6.11. Table 6.12 lists the details on tool allocation and capacity utilization.

For the example (Example 6.3) considered in the Section 6.4.1 with $a_n = 1 \; \forall \; n$ and $b_{ni} = 1 \; \forall \; n$ and $\forall \; i \in B(n)$, the solution for the objective of maximizing the part processing flexibility is given in Table 6.22.

6.5 SUMMARY AND CONCLUSIONS

In the present chapter, certain loading models have been presented which can be used for finding a complete solution to operations assignment problem that could not be obtained while determining the groups of parts and machines using graph partitioning approach discussed in the earlier chapter. These models can also be used in standalone manner for solving operations assignment and tool a cation problems.

In the present chapter, certain issues related with loading problems in FMS environment have been discussed. The issues involved are on the aspects of flexibility and on the problem parameters to be incorporated.

It has been shown that incorporation of flexibility though may make the problem complex to solve and analyze, but may increase the throughput rate, better utilization of resources and the ease at real-time scheduling. Therefore, quantitative measures for the processing flexibility have been suggested and loading models are proposed incorporating such measures. Numerical examples presented exhibit the advantages achieved.

It has been shown that out of tool, pallets jigs and fixtures, important resource to consider is tool with machines for solving the operations assignment problem. The allocation of other resources is proposed to be considered at scheduling stage where allocation decisions will involve number of copies of each

Fazakerley, G.M. (1976), "A Research Report on the Human Aspects of Group Technology and Cellular Manufacture," International Journal of Production Research, Vol. 14, No. 1, 123.

Fisher, M.L. (1981), "The Lagrangian Relaxation Method for Solving Integer Programming Problems," Management Science, Vol. 27, No. 1, 1-18.

Gallagher, C.C. and Knight, W.A. "Group Technology Production Methods in Manufacture," Ellis Horwood Limited, 1986.

Garfinkel, R.S. and Nemhauser, G.L. "Integer Programming," John Wiley & Sons, 1972.

Geoffrion, A.M. (1974), "Lagrangean Relaxation for Integer Programming," Mathematical Programming Study, Vol. 2, 82-114.

Greene, T.J. and Sadowski, R.P. (1987), "A Mixed Integer Program for Loading and Scheduling Multiple Flexible Manufacturing Cell," European Journal of Operations Research, Vol. 24, 379-386.

Groover, M.P. "Automation, Production Systems, and Computer-Integrated Manufacturing," Prentice-Hall Inc., Englewood Cliffs, N.J., USA, 1987.

Guignard, M. and Rosenwein, M.B. (1989), "An Improved Dual-Based Algorithm for the Generalized Assignment Problem," Operations Research, vol. 37, No. 4, 658-663.

Gunasingh, K.R. and Lashkari, R.S. (1989a), "The Cell Formation Problem in Cellular Manufacturing Systems - A Sequential Modeling Approach," Computers and Industrial Engineering, Vol. 16, No. 4, 469-476.

Gunasingh, K.R. and Lashkari, R.S. (1989b), "Machine Grouping Problem in Cellular Manufacturing Systems - An Integer Programming Approach," International Journal of Production Research, Vol. 27, No. 9, 1465-1473.

Ham, I., Hitomi, K. and Yoshida, T. "Group Technology - Applications to Production Management," Kluwer Nijhoff Publishing, 1985.

Harhalakis, G., Nagi, R. and Proth, J.M. (1990), "An Efficient Heuristic in Manufacturing Cell Formation for Group Technology Applications," International Journal of Production Research, Vol. 28, No. 1, 185-198.

Held, M., Wolfe, P. and Crowder, H.P. (1974), "Validation of Subgradient Optimization," Mathematical Programming, Vol. 6, 62-88.

Hyer, N.L. and Wemmerlöv, U. (1989), "Group Technology in the US Manufacturing Industry : A Survey of Current Practices," International Journal of Production Research, Vol. 27, No. 8, 1287-1304.